Application Jukebox Studio

User Guide

Version 8.7

PROPRIETARY AND CONFIDENTIAL TO NUMECENT INC.
CANNOT BE REPRODUCED IN ANY FASHION WITHOUT ADVANCE WRITTEN CONSENT OF THE COMPANY

COPYRIGHT NOTICE

Copyright © 1996 - 2013 Numecent Inc.
15635 Alton Parkway, Suite 100, Irvine, California 92618 USA

All rights reserved.

This document is protected by copyright and contains information proprietary to Numecent Inc. Any copying, adaptation, distribution, or public display of this document without the express written consent of Numecent Inc. is strictly prohibited.

TRADEMARK NOTICE

Numecent, the Numecent logo, Application Jukebox Studio, and the Application Jukebox Studio logo are trademarks or registered trademarks of Numecent Inc.

Microsoft, Windows, Microsoft Office, Microsoft Access, and Visual Studio are registered trademarks of Microsoft Corporation. All other names are used for identification purposes only and are trademarks or registered trademarks of their respective companies.

U.S. PATENT NOTICE

Protected by U.S. Patent Numbers: US 6,453,334; US 6,918,113; US 6,959,320; US 7,043,524; US 7,062,567; US 7,096,253; US 7,240,162; US 7,451,196; US 7,577,751; US 8,024,523
Patents pending in the U.S. and other countries.

Product version: 8.7.1

Guide revision: 1

Last update: October 31, 2013

NOTE: The product version syntax for Application Jukebox Studio is: **M.m.p.bbbbb**
(**M:** major version number, **m:** minor version number, **p:** patch version number, **bbbbb:** build number)

Table of Contents

Introduction **1**

About Application Jukebox Studio 1
Features of Application Jukebox Studio 1
Cloudifying Environment 2
Target Audience 2
Recommended Skills and Knowledge 2
Guide Conventions 3
Getting Support 3

Chapter 1 • Cloudifying Overview **5**

Introduction 5
Data Needed to Cloudify an Application 5
Application Assets 6
Configuration Information 6
The Cloudifying Project 6
Names for a Cloudified Application 6
Core Concepts 7
Virtualization 7
Prefetching 7
Compression 8
Encryption 8
The Jukebox Studio UI 9
The Main Window 9
Workspaces and Views 9
The Project Workspace 11
The AIB Workspace 13
The Token Workspace 13
Other Windows 14
Cloudifying Phases 15
Phase 1: Prepare the Environment 15
Phase 2: Capture the Application Installation 16
Phase 3: Cloudify the Application 16
Phase 4: Test and Optimize the Appset 17
Appset File Structure 19

Chapter 2 • Preparing the Environment **21**

Overview 21

Technical Requirements 21
About Cloudifying Performance 22
Preparing a Clean Cloudifying PC 22
Installing Jukebox Studio 23

Chapter 3 • Capturing the Application Installation 27

Overview 27
Procedure for Capturing the Application Installation 28
Step 1: Create a new Project 28
Step 2: Enter Basic Application Information 28
Application Information 28
Operating System Support 29
Data Compression 30
Data Encryption 31
Other Application Settings 32
Step 3: Start the Application Installation Capture Process 33
Step 4: Install the Application 35
Installation Folder Restrictions 35
Avoiding Artifacts 35
Step 5: Run the application (optional) 36
Step 6: Capturing System Reboots 36
Step 7: Stop the Installation Capture 37
Step 8: Enter Remaining Application Information 38
Revision Information 38
Revision Notes 40
Step 9: Save the Project 40
About OS Compatibility Settings 40

Chapter 4 • Cloudifying the Application 43

Overview 43
Procedure for Cloudifying the Application 43
Step 1: Configure Cloudpaging Settings 43
Step 2: Modify Files, Registry, and Other Assets 44
Step 3: Cloudify the Application 45

Chapter 5 • Testing and Optimizing the Appset 47

Overview 47
Testing the Appset 48
Step 1: Test the appset 48
Step 2: Identify and Solve Problems 48
Optimizing the Appset 49
Step 1: Create Prefetch Files 49
Step 2: Set Compression 52
Step 3: Set Encryption 52
Step 4: Adjust Cache Size 52
Step 5: Cloudify the Application 53
Step 6: Optional: Save the Cloudifying Environment 53
Step 7: Publish the Appset to Jukebox Server 53

Chapter 6 • Cloudpaging Settings 55

Overview 55
Paging Settings 56
Configurable AppEvents Settings 58
Virtualization Settings 62
Sandboxing Settings 63
Security Settings 66
Performance Settings 69

Chapter 7 • Modifying Project Assets 71

Overview 71
Modifying Files and Folders 72
About Folders 73
Root and Template Folders 73
Merged and Non-merged Folders 73
Adding Root Folders 75
Adding Sub-folders 76
Renaming Folders 77
Editing Folder Properties 77
Setting the Working Folder 78
Making Root Folders Merged 79
Adding Files 79
Editing Files 80
Editing File Properties 80
Deleting Files and Folders 81
Excluding Files and Folders 81
Setting the Command Line 82
Locating Files/Folders in Windows Explorer 82
Modifying Registry Entries 83
Adding Registry Keys 84
Renaming Registry Keys 86
Changing the Disposition of a Registry Key 86
Adding Registry Values 87
Editing Registry Values 88
Deleting Registry Keys and Values 88
Excluding Registry Keys 88
Modifying Environment Variables 89
Adding Environment Variables 90
Editing Environment Variables 91
Deleting Environment Variables 91
Excluding Environment Variables 91
Modifying Fonts 92
Adding Fonts 93
Editing Fonts 95
Editing the File Properties of a Font 95
Deleting Fonts 96
Excluding Fonts 96
Locating the File or Registry Entry of a Font 96
Modifying Startup Items 96

Types of Startup Items 97
Adding Startup Items 98
Editing Startup Items 99
Editing the Command Line File Properties of a Start Menu Item 100
Deleting Startup Items 101
Excluding Startup Items 101
Locating the File or Registry Entry of a Startup Item 101
Modifying Services and Drivers 102
Adding Services 103
Editing the Registry Key Properties of a Service 104
Editing the Image File Properties of a Service 104
Editing Services 105
Setting Virtualization Actions for a Service 107
Changing the Failure Action of a Service 107
Deleting Services 107
Excluding Services 108
Locating the Image File or Registry Key of a Service 108
Searching for Assets 109
Finding Assets 109
The Find Results Window 110
About Search Types 111
Using Wildcards 111
Using Regular Expressions 112
Globally Changing Asset Dispositions 113
Understanding Color Coding in Views 114
Conflict Color Coding 115

Chapter 8 • Modifying an Appset 117

Overview 117
Opening an Appset file 118
Editing an AIB file 119
Modifying File Settings 119
Exporting File Information 121
Searching for File Information 121
Modifying Registry Settings 122
Exporting Registry Information 124
Finding Registry Information 124
Modifying Minimum Cache Allocation 125
Editing Virtualization Information 126
Adding Prefetch Files 128
Viewing Prefetch Information 128
Editing Compatibility Settings 129
Editing Token files 130
Modifying an Application Value 130
Modifying Game Rating Settings 132
Modifying OS Compatibility Settings 133
Modifying a Paging Source 134
Modifying a Configurable AppEvent 135
Modifying Sandboxing Settings 138

Modifying Advanced Settings Flags 139
Modifying an Environment Setting 141
Modifying Security Override Settings 143
Saving an Appset File 144

Chapter 9 • Troubleshooting 147

Overview 147
Identifying Common Problems 148
Solutions to Common Problems 148
Missing Shortcuts 148
Blank Shortcuts 149
File Associations Do Not Work 149
Application Fails to Launch 149
Printing Problems 149
Fonts Do Not Appear 150
Folder or Files Disappear from the Virtualized Application 150
Hide "Add/Remove Programs" Entry 150
Roaming and Folder Redirection Issues 150
Prerequisites 151
Unique User Accounts 151
Service Show as Running and Cannot Be Stopped 151
Windows Services or Drivers Do Not Start 152
Plug and Play Device Drivers 153
Alternative Data Streams 154
Anti-virus and Firewall Applications 154
Application Copy Protection 154
Approaches to Uncommon Problems 154

Chapter 10 • Patching an Appset 157

Overview 157
Patching an Appset 158
Modifying a Patched Appset 160
Extracting an Application for Patching 160

Appendix A • Folder Template Mapping 163

Appendix B • Editing Configuration Files 167

Conventions 167
preloadreghives.dat 167
procexcluded.dat 167
filefilt.dat 168
regfilt.dat 168
procfilt.dat 168
defprocsel.dat 168
fileexcluded.dat 169
regexcluded.dat 170
regdifferences.dat 170
rlfnfilt.dat 170
appratng.dat 170

Appendix C • Command Line Usage 171

Glossary 173

Index 183

Introduction

This chapter includes the following topics:

- **About Application Jukebox Studio**
- **Features of Application Jukebox Studio**
- **Cloudifying Environment**
- **Target Audience**
- **Recommended Skills and Knowledge**
- **Guide Conventions**
- **Getting Support**

This guide describes how to use Application Jukebox Studio to transform an application into Application Jukebox cloudpaging format. Additionally, it provides a description of the test environment for cloudified applications.

About Application Jukebox Studio

Application Jukebox Studio is a software tool that encodes Microsoft Windows-based PC applications into a cloudified format to utilize the benefits of cloudpaging. All that is required to cloudify a Windows application are two PCs running Windows XP or later and Jukebox Studio to encode (cloudify) the application.

Features of Application Jukebox Studio

- Simple to use application installation monitoring tool, enabling any Microsoft Windows installer to produce a cloudified equivalent in one pass.
- Most advanced virtualized application delivery solution available. Fully customizable, supporting integrated, isolated and combination virtualization.
- Custom application event handles, allowing scripts to be attached to key application triggers, such as activation, start, close, deactivation, etc.
- Local appset execution mode, allowing an appset to be tested without the need to install on the server.

Cloudifying Environment

To transform an application into a cloudified application, at least two PCs are required: a Cloudifying PC and a Target PC. Depending on the operating system requirements of the intended users, the Cloudifying and Target PCs need to have the Windows XP or later operating system installed.

A “clean” PC is needed each time an application is cloudified, having nothing but the following installed: the OS, Jukebox Studio, all the most recent drivers, any additional programs that the target application needs in order to function properly. Also, if a physical PC is used, an *imaging* program (such as *Ghost*® by Symantec®) should also be installed so the PC is restored to a “clean” state for each cloudifying operation. However, if a virtual machine (VM) is used (recommended for efficiency), then a copy of the “clean” VM can be saved and reused for each cloudifying operation.

NOTE

The Cloudifying PC should also have a minimum screen resolution of 1024 x 768 to accommodate the Jukebox Studio interface.

Target Audience

This guide is written for a technician or engineer who will have the role of *Appset Designer*, the person responsible for using Jukebox Studio to change Windows-based applications into a cloudified format.

Recommended Skills and Knowledge

The following list is meant as a guideline for recommended skills and knowledge in Information Technology (IT), PC platforms and Operating Systems (OS) technologies for an *Appset Designer*, the person who uses Jukebox Studio to cloudify applications.

- Working knowledge of Windows platforms XP and later, the differences between the platforms, and the installation of service packs.
- Working knowledge of hard disk partitions and renaming.
- Working knowledge of basic, distributed and networking file system aspects of the Windows OS, including the organization of Program Files, Roaming profiles, User start menus, and special folders.
- An understanding of DLLs, program executables, shared libraries, and command-line parameters.
- Working knowledge and understanding of the Windows registry, including machine profiles, user profiles, and software registry entries.
- Working knowledge of setting up and configuring Windows applications, such as Microsoft Office.
- Understanding of application installers and their packaging (e.g. InstallShield AdminStudio) will be helpful.
- Experience with an automated/centralized networking tool (e.g., Tivoli, Openview, Unicenter).

NOTE

In order to optimize the performance of each cloudified application, it is recommended that an experienced user of the *target application* participate in the cloudifying process.

Guide Conventions

The following typefaces are used throughout this guide:

Bold text	Names of windows, panes (portions of a window), dialog boxes, menus, commands, buttons, predefined folders, and keyboard keys. Examples: Click **OK**. Press **Enter**. Navigate to the **My Documents** folder.
italic text	Indicates references to a chapter or section within this guide or other documentation. Also used to emphasize new terms. Example: Refer to the *Application Jukebox Studio User Guide* for further information.
> symbol	Sequence in which you should select a menu option. Example: "**File > New > Document**" means "click the **File** menu, click **New**, and then click **Document**."
1, 2, 3	Numbered items indicate sequential steps in a procedure. Sub steps are indicated with indented bullets. Relevant graphics or screen captures may follow steps.
Screen captures	The screen captures in this guide are examples only. They may not exactly match the user interface on your system.

The following icons are used throughout this guide:

NOTE

This icon identifies additional important information relating to information preceding it.

CAUTION

This icon identifies information about actions that need to be taken to prevent possible problems with the application, your local PC system, or your data, including data loss.

Getting Support

Numecent provides a number of convenient ways to obtain technical support as described below.

Web Site

Support information is available online on the following Numecent web site:

http://www.numecent.com/customers/support.html.

Email

You can request technical support by sending us email to **supportteam@numecent.com**. Our support staff will address your email questions in a timely manner.

Telephone

You can obtain telephone support by calling 1 (949) 833-2800 Monday through Friday between the hours of 9 a.m. and 6 p.m. Pacific time (PST).

After Hours Support

For support outside of regular business hours (Monday through Friday, 9 a.m. to 6 p.m. Pacific time) you can contact Numecent via email. Email received outside of the hours noted above will be addressed on the next business day.

Sales

For sales or related information about Application Jukebox Studio, visit our web site (**www.numecent.com**), call us at 1 (949) 833-2800, or send email to **sales@numecent.com**.

Chapter 1

Cloudifying Overview

This chapter includes the following topics:

- **Introduction**
- **Data Needed to Cloudify an Application**
- **Names for a Cloudified Application**
- **Core Concepts**
- **The Jukebox Studio UI**
- **Cloudifying Phases**
- **Appset File Structure**

Introduction

Application Jukebox Studio ("Jukebox Studio" from here on) is used to *cloudify* applications, the process of converting a software application to a form that can be paged to Jukebox Player on users' PCs. To cloudify an application (the *target application)*, it is first installed on a platform that is dedicated to cloudifying and has Jukebox Studio installed. Jukebox Studio determines how the installed application works in the Windows environment and packages the required components and configuration information into a compressed, encrypted file called an *appset*. The appset contains all the files, folders, and registry settings needed to reproduce the application on the user's PC. The appset is loaded on an Application Jukebox Server at which point the target application becomes available for users.

Variations among applications and installation can sometimes make cloudifying seem complex. For example, many applications integrate themselves deeply into the Windows operating system environment, or integrate with other applications. Software vendors also use different installation programs such as InstallShield, InstallAnywhere, etc. To reduce complexity, this guide presents a best practice approach to help ensure smooth and successful cloudifying.

Data Needed to Cloudify an Application

An application is made up of various components such as files and registry settings called *assets*. To cloudify an application, it is necessary to collect and store all assets needed to run the application, along with configuration information that describes how those assets will be processed and managed when paged to Jukebox Player on users' PCs. Jukebox Studio stores the assets and configuration information in a *cloudifying project* (or simply *project*).

The following sections discuss assets, configuration information, and projects in greater detail.

Application Assets

Assets are items needed to cloudify an application. An application may have various types of *assets*. The most common are files, registry entries, and environment variables. There are also more specialized types of assets such as fonts, startup items, services, and drivers that are themselves made up of one or more file and registry entries.

The assets of an application are collected by Jukebox Studio into a project during application installation. The appset designer can later add more assets to the project or remove some, as needed.

Configuration Information

In addition to storing an application's assets, it is also necessary to keep configuration information that describes the application and determines how its assets will be processed and managed when paged to Jukebox Player on users' PCs.

The appset designer makes configuration choices based on the anticipated way that an application will be used, or based on special application needs.

The Cloudifying Project

To cloudify an application, the appset designer must first create a *cloudifying project* (or simply *project*) that will contain all assets and configuration information needed to produce the *cloudified application*, or *appset*. The project contents are saved to a *project file* (**.stw**), also called an *STW file*, for later retrieval.

Assets are added to the project through the application *capture* process. During the capture process, the application installer is run and Jukebox Studio monitors changes made to the system. Specifically, it keeps track of files and registry values that are added to the system or modified, along with their folder and registry key locations. At the end of the process, Jukebox Studio adds to the project the tracked assets that can then be fine-tuned by the appset designer, along with the configuration information, if needed.

Finally, the project is used to produce an appset through the *cloudifying* process.

Names for a Cloudified Application

There are three different names for a "cloudified application" depending on the point of view (POV) of the person using it:

- Appset Designer's POV - "**appset**"
- Jukebox Server Admin's POV - "**application**"
- User's POV - "**cloudified application**"

  After a cloudified application has been paged, it is referred to as a "**paged application**."

Core Concepts

Virtualization

Application virtualization allows for the application to be paged to a user's PC while protecting the operating system from resource conflicts and environment corruption that can occur with traditional installation methods. Virtualization creates a boundary between the application resources and the local system resources. Application Jukebox Studio takes virtualization to a new level by making virtualization configurable.

Virtualization is an Application Jukebox Studio feature that makes it possible to assign a specific disposition layer to individual application *assets*, such as files, folders, registry keys, and registry values. The layer determines whether the system resource can be seen by the local system, and whether it is permanent or can be removed.

For certain resources, such as fonts and environment variables that are typically shared by the entire system, a special method called *injection* is used to maintain the isolation layer. There are four *Disposition Layers*:

- **Installed-permanent (Layer 1)** - Copies assets (files, folders, registry keys, and registry values) permanently onto the local system, and can be seen by the entire local system.
- **Installed-temporary (Layer 2)** - Installs assets during the activation process, and uninstalls assets during the deactivation process. If the asset already exists on the local system, the original asset is backed up before the new asset is installed, and when the new asset is uninstalled the original asset is restored.
- **Virtual-integrated (Layer 3)** - **(default setting)** Assets that can be seen both by the paged application and the local system, but are not physically installed on the local system.
- **Virtual-isolated (Layer 4)** - Assets that can only be seen by the paged application, and are not physically installed on the local system.

Prefetching

Prefetching is the process of fetching application pages prior to their being explicitly requested by Jukebox Player, in anticipation of being needed by the application. Unlike conventionally installed applications, application paging does not require all application files to be always available. Instead, only some key pages from the ASP's server are necessary for the application to launch and perform basic functions.

The prefetch file is created by logging pages that are being fetched when the application is executed. This includes all the functions required by the application when the user opens it for the first time. During the optimization phase of the cloudifying process, you can capture three types of Prefetch files:

- **Stage-1 Prefetch** - This group of application pages are those that are needed to start the application, as well as to perform typical actions for the first few minutes of using the program, such as opening files, saving files, etc. The process of fetching these pages from the server to Jukebox Player always happens before an application launches. **This stage is always recommended for best application performance.**
- **Stage-2 Prefetch** - This group of data consists of application pages that are needed to perform other important tasks the user is likely to perform. The amount of captured Stage-2 Prefetch pages depends on the trade-off among several factors such as the time it will take to fetch these pages, the cache size required to store them, etc. Examples of Stage-2 Prefetch are using various functions of a business application and loading the second level of a game. Jukebox Player will fetch these pages slowly in the background after the application launches. This stage is optional.

- **Stage-N Prefetch** - You can add Additional prefetches for specific events in an application. These work on a page-based trigger to fetch an additional set of pages for the application just before it would on-demand fetch the pages. For example, if a game needs to load the next level, then this level could be recorded into a stage-3 prefetch. The first 10 blocks recorded are used as a trigger to fetch the entire stage. If Jukebox Player needs to on-demand any of the first 10 blocks, this will cause the trigger to fetch that entire stage. There can be any number of these additional stages and they are all optional.

Compression

Compression of an appset decreases the file size, and the smaller it is, the faster the appset pages. Application paging operates on file pages as units of file contents. Each application file to be cloudified is divided into a series of pages stored in a content (STC) file by Jukebox Studio, published to Jukebox Server, and then transmitted by Jukebox Server, received and virtualized on the local PC by Jukebox Player. Since file pages are passed through many components normally residing on different machines, the pages of data should be compressed to make data transmission and storage efficient.

Encryption

Encryption of appsets provides a high degree of security, ensuring that the contents of the appset remains safe and private when paging over the Internet. Each page in the STC file is individually encrypted, which protects the content during transport and when cached. As with compression, since the pages of data can reside on various machines, it is recommended to always encrypt the content.

The Jukebox Studio UI

The Main Window

The Jukebox Studio UI consists of a *main window* containing a *menu bar*, a *toolbar*, and a *status bar*, each of which can be individually hidden. The rest of the window, between the toolbar and status bar, is called the *common area*; it is initially blank when Jukebox Studio starts up.

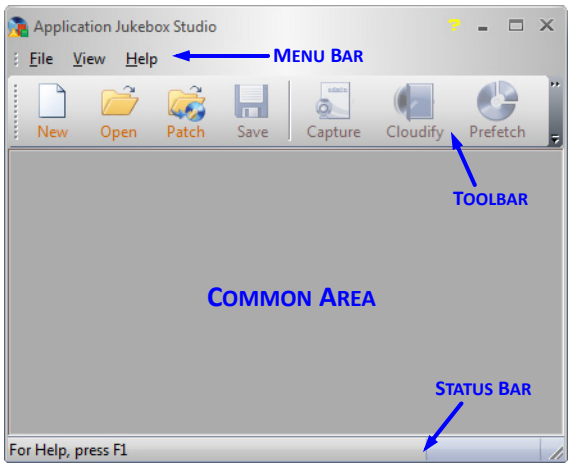


Figure 1-1: Jukebox Studio Main Window at Startup

The toolbar contains general commands that reflect the typical cloudifying workflow.

The status bar displays a brief description of a command when the mouse cursor hovers over a menu item or toolbar command. It also shows the state of the **Caps Lock**, **Num Lock**, and **Scroll Lock** keys. It may show other context-specific information as well.

The common area typically hosts one or more workspaces in a tabbed fashion (see Workspaces and Views on page 9). Each workspace, in turn, hosts one or more views in a tabbed fashion as well. Additionally, the common area may host other special windows such as the **Find Results** window (see Other Windows on page 14).

Workspaces and Views

When the appset designer creates a new project or opens an existing project file, a *workspace* is created for the project. The workspace is the area of the main window where the appset designer can configure the project's settings, manage its assets, and cloudify an application. By default it occupies the entire common area of the main window.

NOTE

There are other types of workspaces that are associated with the different types of files that Jukebox Studio can open. In addition to project files, Jukebox Studio allows opening *AIB files* and *token files*, which are appset-related files discussed in Appset File Structure on page 19. These files get opened in an *AIB workspace* or *token workspace*, respectively (see The AIB Workspace on page 13 and The Token Workspace on page 13).

Multiple workspaces (possibly of different types) can be open simultaneously. Workspaces are tabbed at the top to make it easy to switch between them by clicking their corresponding tabs. The workspace where the appset designer is currently working is the *active workspace*.

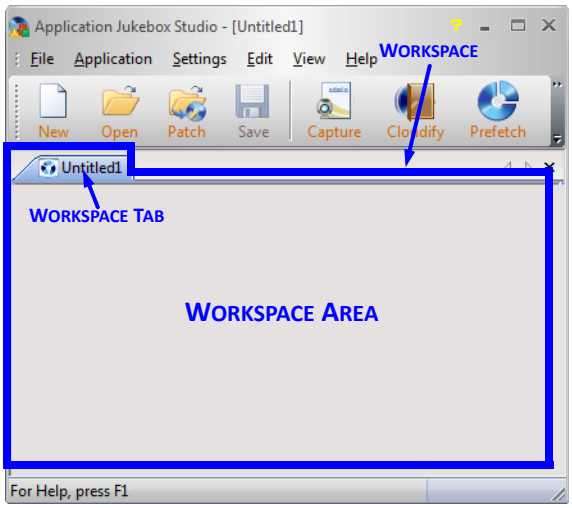


Figure 1-2: Jukebox Studio Workspace

A workspace is made up of one or more *views* depending on its type. Views are used to organize a workspace to make it easier for an appset designer to manage its contents. Each view is used to manage a different type of content or, sometimes, to present the same content in different ways. Views are tabbed at the bottom within the workspace area so it is easy to switch between them by clicking their corresponding tabs. The view that is currently selected is the *active view* and it occupies all the available workspace area.

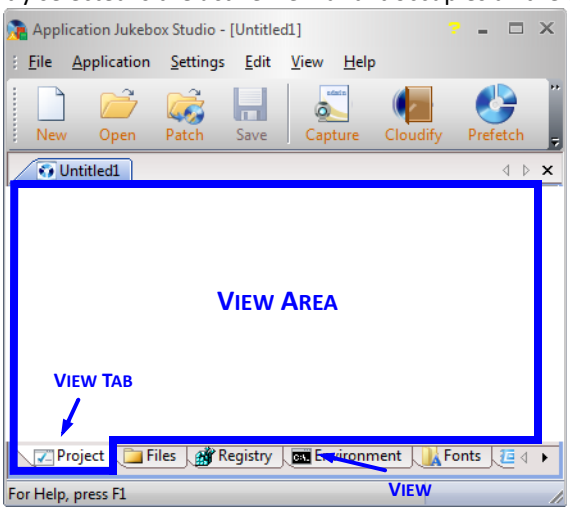


Figure 1-3: Jukebox Studio View

The figure below shows the main window with an open project:

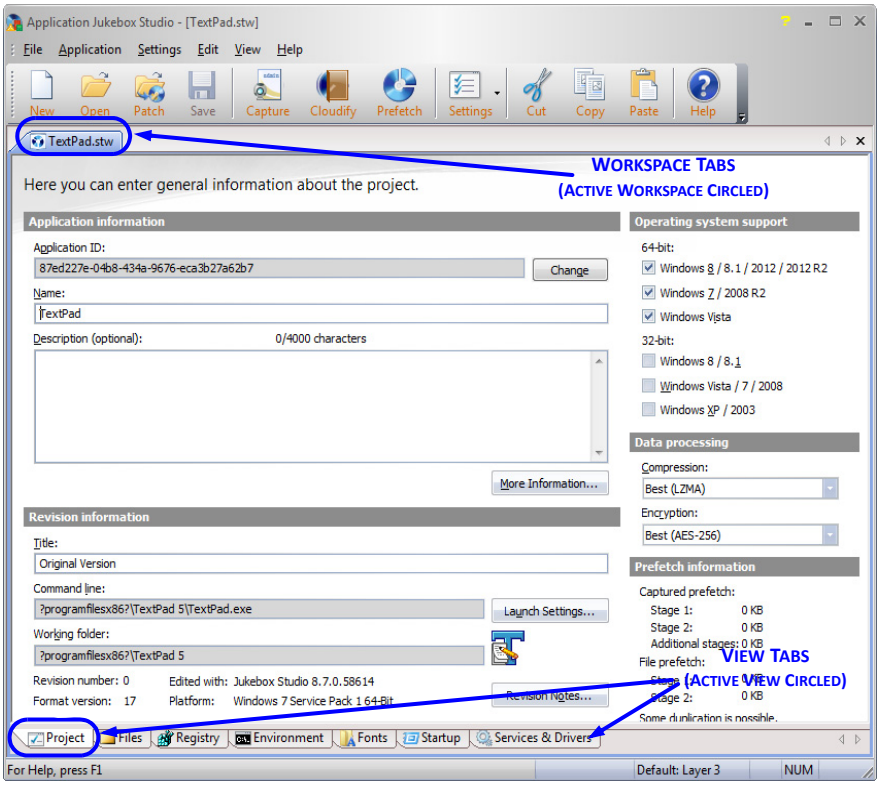


Figure 1-4: Main Window with an Active Project Workspace

The following sections discuss each type of workspace and the views they contain.

The Project Workspace

Jukebox Studio creates a project workspace when a new cloudifying project is created or an existing project file is opened. It contains the following views:

- *Project view* (see Enter Basic Application Information on page 28)
- *Files view* (see Modifying Files and Folders on page 72)
- *Registry view* (see Modifying Registry Entries on page 83)
- *Environment view* (see Modifying Environment Variables on page 89)
- *Fonts view* (see Modifying Fonts on page 92)
- *Startup view* (see Modifying Startup Items on page 96)
- *Services & Drivers view* (see Modifying Services and Drivers on page 102)

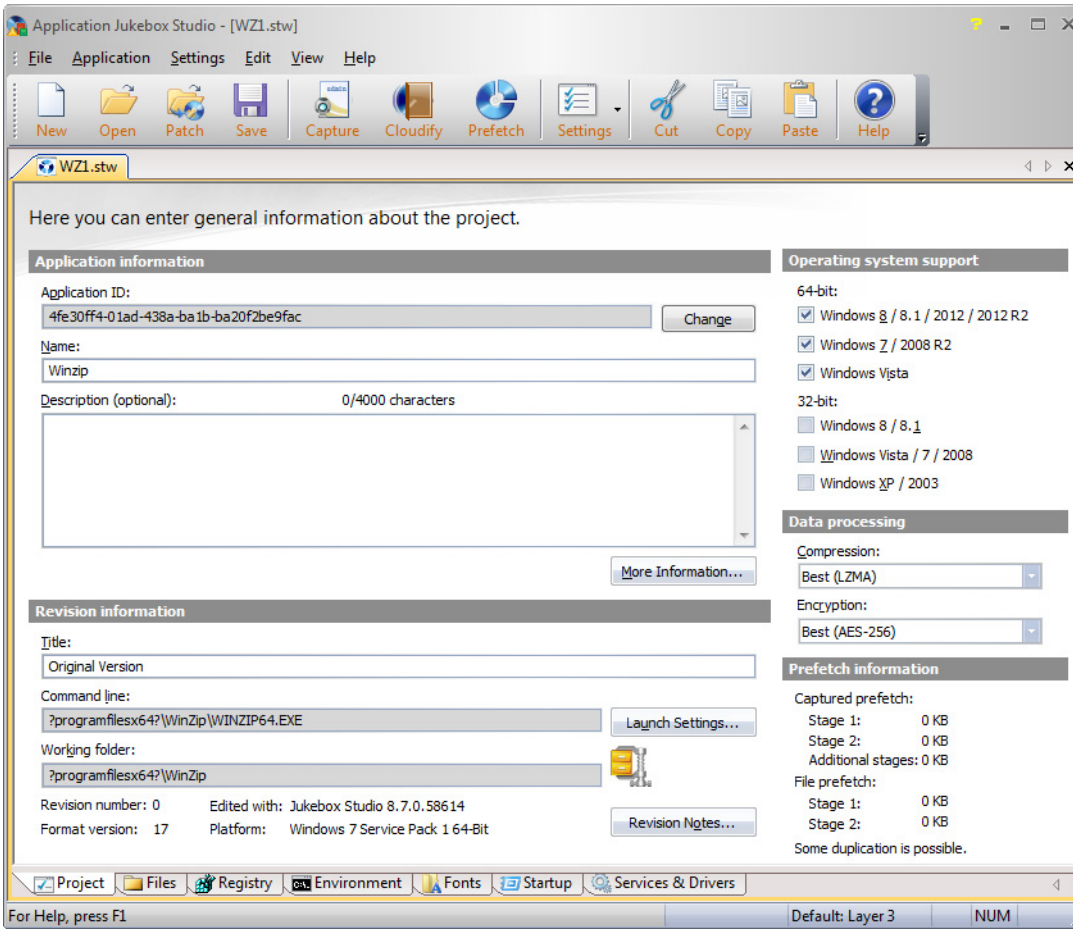


Figure 1-5: Project Workspace and Views

The AIB Workspace

An AIB workspace is created when an existing AIB file is opened for editing (see Editing an AIB file on page 119). It contains a single *AIB view* as shown in the figure below.

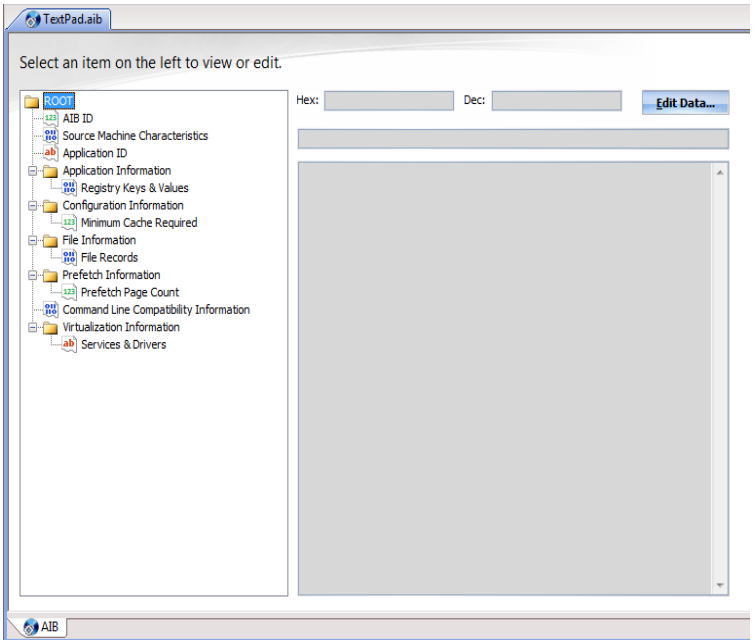


Figure 1-6: AIB Workspace and View

The Token Workspace

An token workspace is created when an existing AIB file is opened for editing (see Editing Token files on page 130). It contains a single *Token view* as shown in the figure below.

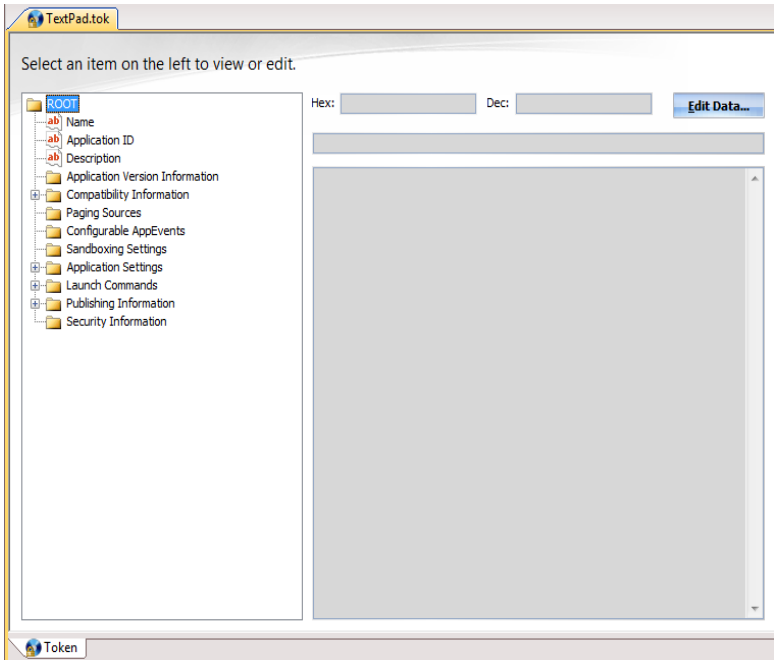


Figure 1-7: Token Workspace and View

Other Windows

A workspace may contain additional special-purpose windows that appear alongside the views in the workspace area, such as the **Find Results** window (project workspace only). However contrary to views, which are always visible, these special windows may be shown or hidden as desired.

Other windows called *dialogs* appear on top of the main window when certain commands are invoked to obtain user input. Most Jukebox Studio dialogs are *modal* and must be dismissed before it is possible to interact with the main window again. Others (such as the **Find** dialog) are *modeless* and allow switching between them and the main window.

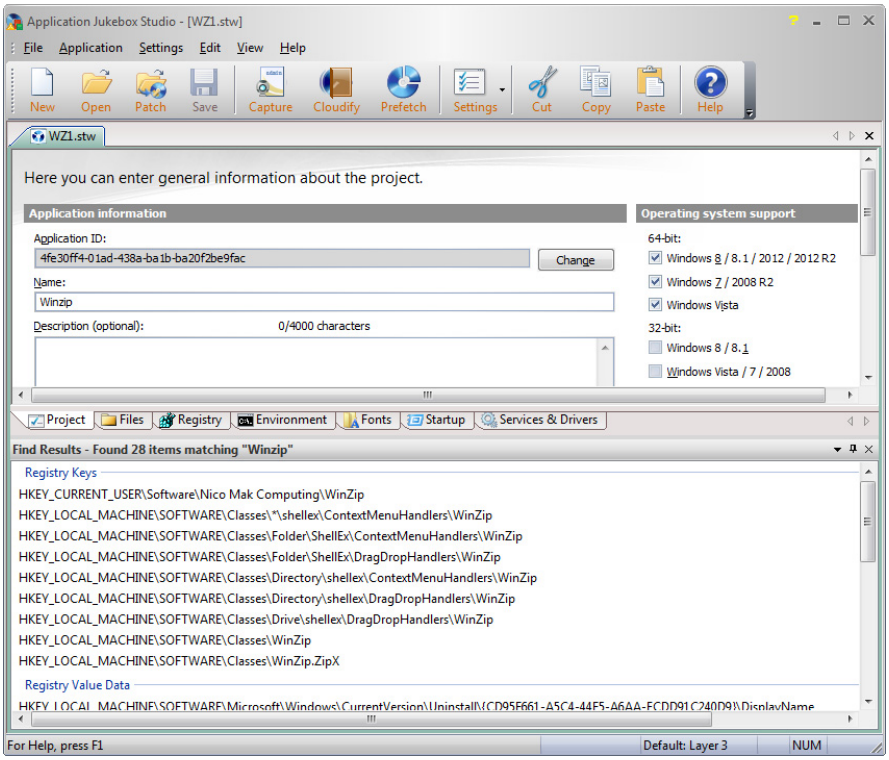


Figure 1-8: Find Results Window

Cloudifying Phases

As illustrated below, there are **four phases** that are required to cloudify an application using Application Jukebox Studio. Each cloudified application can later be added to a Jukebox Server and paged to Jukebox Player installed on users' PCs.

This section presents an overview of the four cloudifying phases. Each phase is covered in detail in a subsequent chapter.

Figure 1-9: Four Phases of the Cloudifying Process

Phase 1: Prepare the Environment

See **Chapter 2**, Preparing the Environment on page 21.

Cloudifying applications with Application Jukebox Studio requires at least two (physical or virtual) PCs: a *Cloudifying PC* with Jukebox Studio installed, and one or more *Target PCs* with Jukebox Player installed, for testing. The desired operating system (Windows XP or later, or Windows 2003 server or later), all necessary drivers, and any prerequisite software (such as the most recent OS service packs or .NET) should be installed on all PCs.

If the appset is intended to run on multiple operating systems, then it is recommended to always cloudify on the earliest OS version. For example, if the application works on both Windows XP and Windows 7, then it is best to cloudify the application on Windows XP. If the application works on Windows 7 32-bit and 64-bit, then it is best to cloudify on Windows 7 32-bit. Cloudifying on the earliest OS typically has the best chance of working because that OS will have less dependencies already installed as opposed to later OS versions. This same general rule applies to OS service packs.

CAUTION

It is not always possible to have a single appset work on all target operating systems. In this case multiple appsets may be required for the different operating systems.

If physical PCs are used for the cloudifying, an imaging utility such as *Ghost* from Symantec should also be installed so all PCs can be easily restored to their pre-cloudifying state. For maximum cloudifying efficiency, however, Virtual Machines (VMs) are recommended.

In order to optimize the performance of each cloudified application, it is recommended that an experienced user of the target application participate in the cloudifying process. Otherwise the appset will still function but maybe not a optimal as would be possible.

Phase 2: Capture the Application Installation

See **Chapter 3**, Capturing the Application Installation on page 27.

During the capture process, the application installer is run and Jukebox Studio monitors system changes (files and registry values that are added to the system or modified). Each time you capture the installation process of an application, it is necessary to start with a Cloudifying PC and Target PCs that are in their pre-cloudifying state, as described in Phase 1. If the capture process is interrupted for any reason (such as a system reboot after the installation is complete) the project file allows the capture process to resume.

To capture the installation of an application, do the following:

Step 1: Create a New Project - Click **New** on the Jukebox Studio toolbar. The Project view of the UI appears.

Step 2: Enter Application and Publisher Information - From the Project view of the Jukebox Studio UI, enter the following information (often provided on software product packaging):

- Application Information
- Application Description
- System Requirements
- Operating System Support

Step 3: Create an Install Folder - Create a folder on the C drive of the Cloudifying PC (e.g., C: \Program Files\MSWord2007).

Step 4: Start Application Capture - Click **Capture** on the Jukebox Studio toolbar. The **Capture in Progress** dialog appears. Leave this dialog open. Later when the application installation process has completed, you will click **OK** to end the capture process.

Step 5: Install the Application - Install all components of the target application using the **Custom** installation option provided by most installation programs. This is advised so that all application components will be available to users of varying needs.

Step 6: Run the application (optional) - The purpose for running the application at this point is to capture settings (such as registry modifications) that are made after the installation is completed. Such cases are processes such as displaying the EULA (End User License Agreement) or changing user profile settings. After the installation is complete, it is advised to run the application and try using some menu options and features to verify that related components are properly installed.

Step 7: Stop Application Capture - Click **OK** on the Capture in Progress dialog.

Step 8: Save the Project - Click **Save** on the Jukebox Studio toolbar.

Phase 3: Cloudify the Application

See **Chapter 4**, Cloudifying the Application on page 43.

Step 1: Configure project settings - From the Project view and Settings dialog of the Jukebox Studio UI, enter the following project information include:

- Application Information
- Supported Operating Systems
- Data Compression Method
- Data Encryption Method
- Paging Settings
- Configurable AppEvents Settings
- Virtualization Settings
- Sandboxing Settings
- Security Settings
- Performance Settings

Step 2: Modify File, Registry, and Environment Content and Settings - As required by the particular application, you can modify these Files, Registry, and Environment views of the Jukebox Studio UI.

NOTE

For complete instructions see **Chapter 7**, Modifying Project Assets on page 71.

Step 3: Cloudify the Application - Click **Cloudify** on the toolbar.

After the cloudifying process is complete, the result is an ***appset***. The cloudified application is called an *appset* from the point of view (POV) of an *Appset Designer*, the person who uses Jukebox Studio to change Windows-based applications into a cloudified format. (A cloudified application is called a *cloudified application* from the user's POV, and an *application* from the Jukebox Server admin' POV).

Strictly speaking, the appset (**.stp** file) is the outer layer that ecapsulates the other three application components: the STC file (**.stc** - the code and other configuration information that comprises most of the application), the AIB file (**.aib** - metadata that describes the content of the STC file), and the **Token file** (comprised of security data). Later, after the final appset is created, it is then *published* to the Jukebox Server and becomes available for cloudpaging to users.

Phase 4: Test and Optimize the Appset

See **Chapter 5**, Testing and Optimizing the Appset on page 47.

Testing the Appset

Step 1: Test the Appset. Testing the appset includes loading it onto a Target PC with Jukebox Player installed and running the application. *Local paging*, the process of paging the appset to Jukebox Player on a single Target PC, allows you to test several appset iterations quickly, without publishing the appset to Jukebox Server

Step 2: Identify and Solve Problems. While it is common to encounter few if any problems while testing an appset, it is often necessary to solve problems. Procedures are provided to identify problems and solve them.

Optimizing the Appset

NOTE

It is necessary to test the appset after completing each of Steps 1-4 below, and then make configuration adjustments as required.

Step 1: Create Prefetch Files. After installing Jukebox Player on the Target PC, you will need to enable the prefetch Capture button on the Jukebox Player toolbar (explained later), which includes Stage-1 and Stage-2 options. With Stage-1 selected, you will then click **Capture** to start capturing the Stage-1 prefetch, perform the desired actions on the target application, and click **Capture** to end the capture process. Then, if you choose to add a Stage-2 prefetch, you will select Stage-2 and repeat the same process.

Step 2: Set Compression. During the testing phase, compression is usually set at “None” or “Fastest” in order to speed up the cloudifying process. After testing is complete, you can change compression to “Best” for the optimal delivery speed.

Step 3: Set Encryption. Again, during the testing phase, encryption is usually set at “None” or “Fastest” in order to speed up the cloudifying process. After testing is complete, you can change encryption to “Best” for the optimal security.

Step 4: Increase the Cache Size. (Optional) Some applications may perform better with a larger Jukebox Player cache. This cache setting will override Jukebox Player’s cache setting set by the user. The cache size is limited by the available hard disk space on the user’s PC.

Step 5: Save the Project file. This will be required to patch the appset in the future.

Step 6: Publish the appset to Jukebox Server. Once optimized, the appset is ready to be *published* from Jukebox Studio to the Jukebox Server, which is done by copying the final STC file to the Jukebox Server. After the Jukebox Server admin completes necessary server configurations, the application is ready to be paged to the Jukebox Player running on users’ PCs.

Appset File Structure

An appset is made up of multiple files. The primary files described below are always present whereas the additional files may or may not be in the appset depending on the configuration settings of the project used to create the appset. All file types are shown with their corresponding file extensions in parentheses.

Primary files

STP or ***appset file*** (.stp)

Compressed file that contains all of the other files below. It can be uncompressed by changing the extension to **.zip** and unzipping the file.

STC (.stc)

Contains all the data pages of the application being cloudified. This file is the largest portion of the STP file.

AIB (.aib)

Contains all the metadata for application files and registry keys.

TOK or ***token file*** (.tok)

Contains the security data and general information about the application.

Additional files

ICO or ***icon file*** (.ico)

Contains the graphic file for the application icon provided by the software vendor (example, Word icon for each Word file).

CAE or ***Configurable AppEvent file*** (.cae)

There is one **.cae** file per AppEvent trigger and each can initiate an action such as running a VB script that installs other needed applications.

HTML (.html)

End User License Agreement (EULA) for the application.

Chapter 2

Preparing the Environment

This chapter covers these topics:

- Overview
- Technical Requirements
- Preparing a Clean Cloudifying PC
- Installing Jukebox Studio

Overview

As described in *Cloudifying Overview* chapter, application cloudifying with Jukebox Studio has four phases. **Phase 1: Preparing the Environment** includes procedures for preparing a clean cloudifying PC, preparing the testing PC(s), and the procedure for installing Jukebox Studio.

Figure 2-1: Phase 1 of the Cloudifying Process

Technical Requirements

Two or more PCs -You will need one Windows-based PC dedicated to cloudifying applications (*Cloudifying PC*) and one or more Windows-based PCs to test the cloudified applications (*Target PCs*).

The Cloudifying PC must meet or exceed the system requirements of the application being cloudified.

- **Operating System** - One of the following:
  - Windows XP with Service Pack 3 or greater (32-bit)
  - Windows Vista with Service Pack 2 or greater (32-bit or 64-bit)
  - Windows 7 (32-bit or 64-bit)
  - Windows 8 (32-bit or 64-bit)
  - Windows 8.1 (32-bit or 64-bit)
  - Windows Server 2003 R2 (32-bit)
  - Windows Server 2008 (32-bit)

  - Windows Server 2008 R2 (64-bit)
  - Windows Server 2012 (64-bit)
  - Windows Server 2012 R2 (64-bit)
- **Hardware** - The following requirements have been established based on acceptable compression and encryption performance:
  - **CPU:** 2.0 GHz or faster
  - **RAM:** 1GB or greater
  - **Disk Space:** 40 GB or greater (estimated), depending on the installation size of the application to be processed.
- **Cloudifying Software** - Application Jukebox Studio. An "administrator" privilege level is required to install and create appsets, although not to modify existing appsets.
- **Recommended Software** - Virtual Machine software (such as *Virtual PC*® from Microsoft) or, for physical machines, Disk Imaging software (such as *Ghost*® from Symantec®)

  For the Target PCs:
- **Test Application Software** - Application Jukebox Player (Please refer to the *Application Jukebox Player User Guide* for additional requirements).
- **Support Software** - As required by the paged application. Depending on the application, specific video card drivers (e.g. DirectX) may be required. See the application manufacturer's website for current updates.
- **Additional Software** - Any other software as required to best reflect the typical system for targeted users. For example, Microsoft Office for users in an office environment.

About Cloudifying Performance

The cloudifying process is both I/O and CPU intensive. Performance measurements have shown that the following criteria have the greatest effect on cloudifying speed, in increasing order:

- Physical hardware (>30% speed improvement over a virtual machine)*
- SCSI instead of IDE hard drives
- Faster front side bus
- Faster processor
- Dual/Quad processor instead of single processor (>10% speed improvement)*

* These measurements are provided only as a reference. Actual results may vary based on your hardware and software system configuration.

NOTE

PCs can be physical computers or virtual machines (VMs). While physical machines are faster, the overall cloudifying process is quicker using VMs.

Preparing a Clean Cloudifying PC

A "clean" PC is needed to cloudify an application (*Target Application*). As listed in *Technical Requirements* above, a clean PC consists of the OS, Application Jukebox Studio, and for physical PCs, an imaging program like Symantec's Ghost™.

Uninstalling extraneous applications is typically not sufficient, since unwanted *artifacts* (DLLs, etc.) are often left behind. Not removing other applications and artifacts could cause conflicts during the cloudifying process, cause extra data to be included, and cause other conditions that prevent the process from functioning correctly.

To prepare a clean Cloudifying PC, do the following:

1. Format the disk drive and install the Windows XP or later operating system. **Make sure that you have the manufacturer's Recovery Disk or Original Driver Disk when doing this step.**
2. Install any needed drivers and the latest OS service packs. Driver discs for various system components such as the network card, video drivers, etc. should come with the PC. However, you may have to download the drivers from the Internet or your company server.
3. Install Jukebox Studio on the cloudifying PC. See the *Installing Jukebox Studio* in this chapter for complete instructions.
4. Create a duplicate of the clean PC.
   - **For a Virtual Machine:** Take a snapshot of the VM, which can later be reverted or cloned.
   - **For a physical PC:** Save an image of the Cloudifying PC configuration using a disk imaging utility such as Symantec's Ghost™. This process consists of copying the entire PC configuration to a separate or disk partition for later reconfiguration of the same or another PC.

CAUTION

Jukebox Studio can only capture assets being installed to the same hard drive on the Cloudifying PC where Windows is installed. It can be any drive letter, but both the capture assets and Windows must reside on the same drive.

NOTE

Virtual PCs (VMs) are recommended for both the Cloudifying PC and Target PCs over physical PCs, because a clean VM can be copied and therefore can provide the fastest recovery of a clean OS. While a physical PC can be faster for the last cloudifying phase, virtual PCs are faster for the initial phases, and are faster for the cloudifying process as a whole.

Installing Jukebox Studio

The Application Jukebox Studio installer is provided to you either by CD or email. After you have completed the environment preparation, follow these steps:

To install Jukebox Studio:

1. Run the Jukebox Studio installer: **Application Jukebox Studio 2008.msi**.

The **Application Jukebox Studio - InstallShield Wizard** dialog appears.

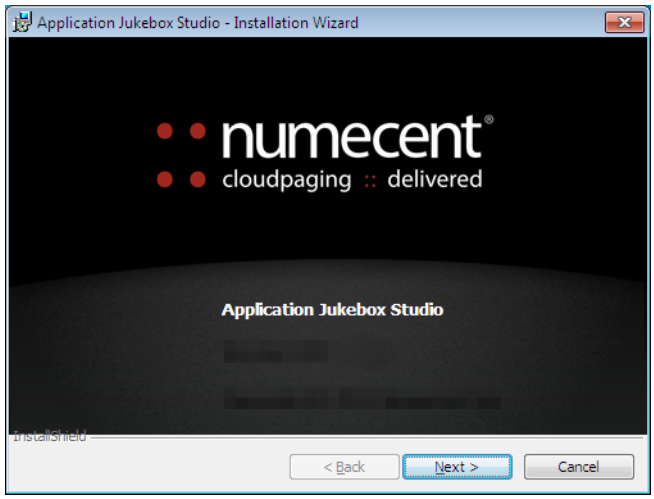


Figure 2-2: Installer Setup Dialog

2. Click **Next**. The **License Agreement** dialog appears.

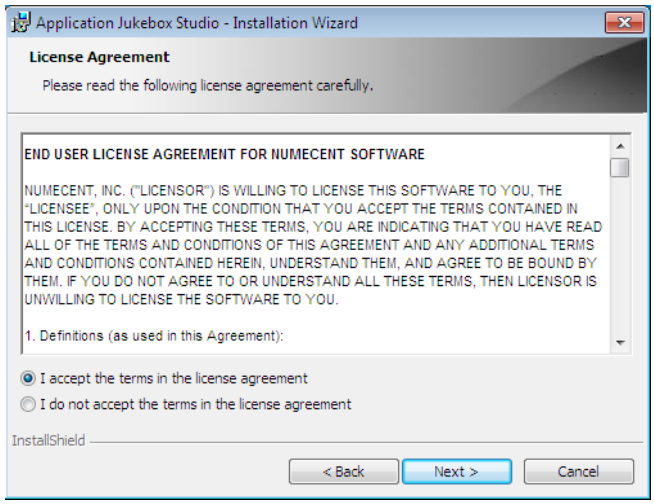

Application Jukebox Studio - Installation Wizard
License Agreement
Please read the following license agreement carefully.
END USER LICENSE AGREEMENT FOR NUMECENT SOFTWARE
NUMECENT, INC. ("LICENSOR") IS WILLING TO LICENSE THIS SOFTWARE TO YOU, THE "LICENSEE", ONLY UPON THE CONDITION THAT YOU ACCEPT THE TERMS CONTAINED IN THIS LICENSE. BY ACCEPTING THESE TERMS, YOU ARE INDICATING THAT YOU HAVE READ ALL OF THE TERMS AND CONDITIONS OF THIS AGREEMENT AND ANY ADDITIONAL TERMS AND CONDITIONS CONTAINED HEREIN, UNDERSTAND THEM, AND AGREE TO BE BOUND BY THEM. IF YOU DO NOT AGREE TO OR UNDERSTAND ALL THESE TERMS, THEN LICENSOR IS UNWILLING TO LICENSE THE SOFTWARE TO YOU.
1. Definitions (as used in this Agreement):
I accept the terms in the license agreement
I do not accept the terms in the license agreement
InstallShield
< Back
Next >
Cancel

Figure 2-3: License Agreement Dialog

3. Read the license agreement, select **I accept the terms in the license agreement**, and click **Next**. Otherwise, click **Cancel** to cancel the installation.

The **Destination Folder** dialog appears.

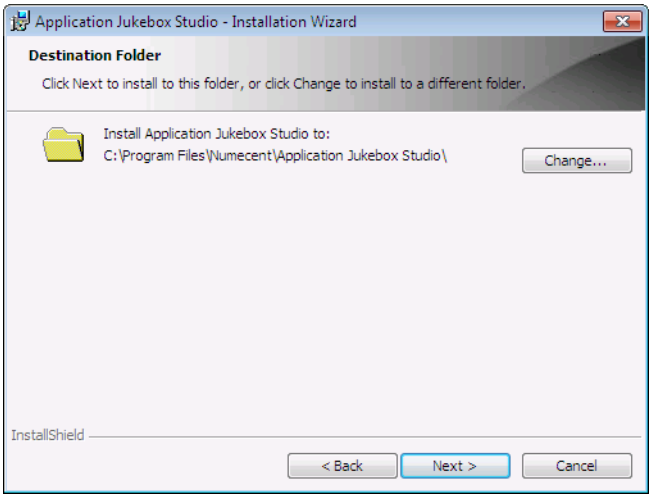

Application Jukebox Studio - Installation Wizard
Destination Folder
Click Next to install to this folder, or click Change to install to a different folder.
Install Application Jukebox Studio to:
C:\Program Files\Numecent\Application Jukebox Studio\
Change...
InstallShield
< Back
Next >
Cancel

Figure 2-4: Destination Folder Dialog

4. Specify the directory into which you want to install Jukebox Studio and click **Next**, or click **Back** if you want to change previous settings.

   The **Ready to Install the Program** dialog appears.

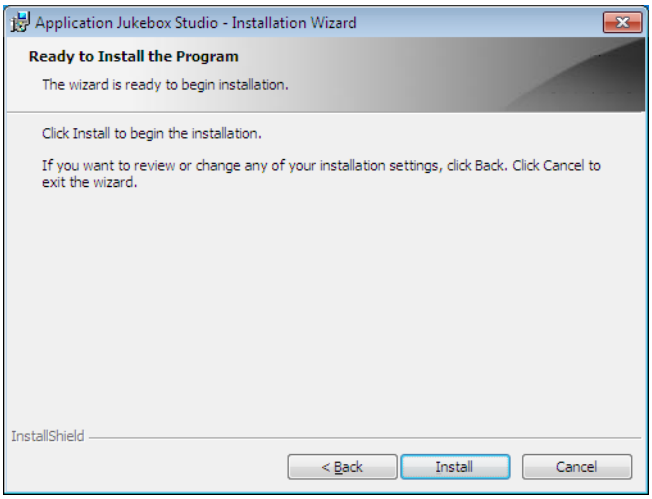

Application Jukebox Studio - Installation Wizard
Ready to Install the Program
The wizard is ready to begin installation.
Click Install to begin the installation.
If you want to review or change any of your installation settings, click Back. Click Cancel to exit the wizard.
InstallShield
< Back
Install
Cancel

Figure 2-5: Ready to Install the Program Dialog

5. Click **Install**, and then click **Next** on the subsequent dialogs.

The **Installing Application Jukebox Studio** dialog appears.

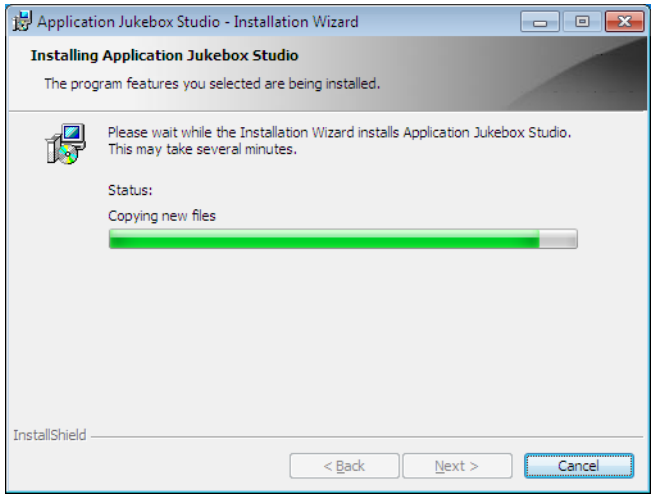

Application Jukebox Studio - Installation Wizard
Installing Application Jukebox Studio
The program features you selected are being installed.
Please wait while the Installation Wizard installs Application Jukebox Studio. This may take several minutes.
Status:
Copying new files
InstallShield
< Back
Next >
Cancel

Figure 2-6: Installing Application Jukebox Studio Dialog

6. Click **Finish**. The installation has completed.
   The **Installation Wizard Completed** dialog appears.

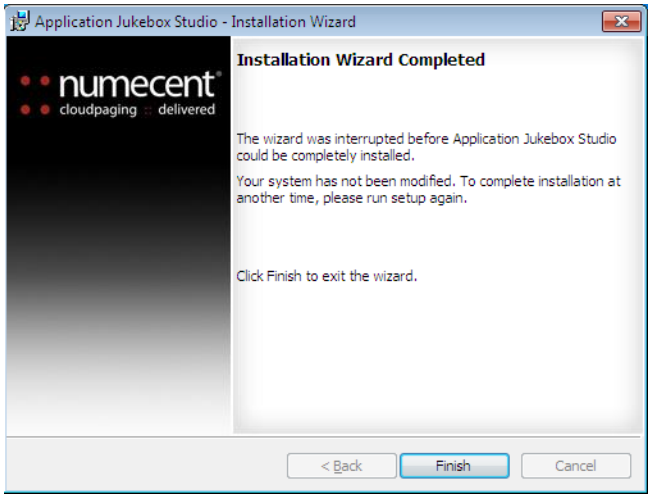

Application Jukebox Studio - Installation Wizard
numecent
cloudpaging delivered
Installation Wizard Completed
The wizard was interrupted before Application Jukebox Studio could be completely installed.
Your system has not been modified. To complete installation at another time, please run setup again.
Click Finish to exit the wizard.
< Back
Finish
Cancel

Figure 2-7: Installation Wizard Completed Dialog

Chapter 3

Capturing the Application Installation

This chapter covers these topics:

- **Overview**
- **Procedure for Capturing the Application Installation**
  1. Create a new Project
  2. Enter Basic Application Information
  3. Start the Application Installation Capture Process
  4. Install the Application
  5. Run the application (optional)
  6. Capturing System Reboots
  7. Stop the Installation Capture
  8. Enter Remaining Application Information
  9. Save the Project
- **About OS Compatibility Settings**

Overview

As described in *Cloudifying Overview* chapter, application cloudifying with Jukebox Studio has four phases. **Phase 2: Capturing Installation** includes all the steps pertaining to capturing the installation of the target application and installing any associated applications that are needed for the target application to function properly.

Figure 3-1: Phase 2 of the Cloudifying Process

NOTE

Regarding the capture phase of the cloudifying process, even though the program may require a specific version of DirectX, Acrobat, or QuickTime; these programs/plug-ins are not needed for the processing system. They will need to be installed onto the testing system.

Procedure for Capturing the Application Installation

Step 1: Create a new Project

Click **New** on the Jukebox Studio toolbar. A new project is created that can be saved to a project file (**.stw**). The project file is where all cloudifying configuration settings and assets are stored.

Figure 3-2: New project button on the toolbar

NOTE

It is recommended to clear the Log file for each new project since the log file is appended to each time. This can accomplished by clicking the **Application** menu and then select **Clear Log**.

Step 2: Enter Basic Application Information

On the **Project** view of the Jukebox Studio UI, enter the following information, which is typically printed on the software cloudifying or in other materials provided by the publisher:

Application Information, **Application Description**, **System Requirements**, and **Operating System Support**.

Application Information

- **Application ID** – This is a unique identifier that is newly generated each time you click **Change**.
- **Name** – This name is displayed in the Jukebox Player Application pane.
- **Application Description** – This field describes the capabilities and uses for the software program. This description is used primarily for informational purposes and can be shown on the Application Jukebox Studio server.

NOTE

Game-specific information: If you are cloudifying a game, you can include a game rating (ESRB) code so parents can determine if the game is age-appropriate for their children. For more information see Virtualization Settings on page 62.

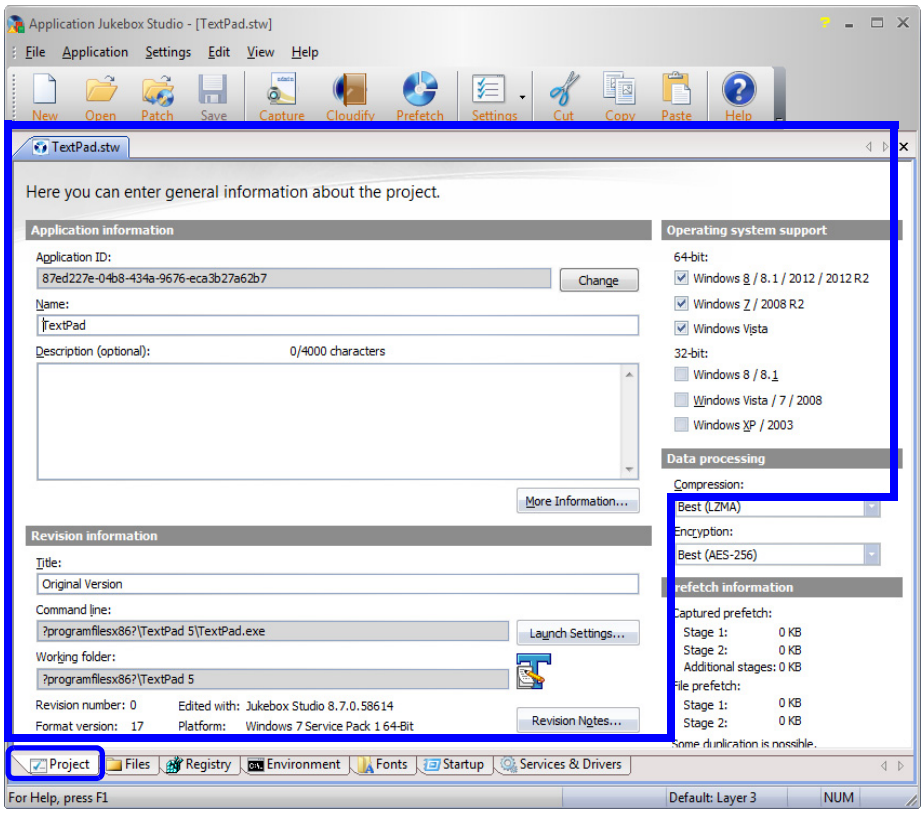


Figure 3-3: Project-View - Basic Project Information & Settings

Operating System Support

The Operating System Support dialog specifies the Operating Systems for which the application is compatible. Jukebox Player will use this information to determine if the appset is compatible with user's Operating System. The Operating System compatibility information is usually included with the product information printed on the application box or on the publisher's web site, or is determined through testing.

NOTE

If multiple OS version are desired as the Target OS, then it is recommended to always cloudify on the earliest version. For example, if an application support Windows XP and Windows 7, then it is best to cloudify on XP. The same is recommended for service packs.

If the processed application is run on a machine for an unsupported Operating System, Jukebox Player will display a dialog indicating that the Operating System is not supported and will not start the application. Sometimes testing indicates that an application is compatible with an Operating System not specified in the product literature.

Selecting the OS options sets the default operating system(s) for which the target application is compatible. The Admin service has the capability to override the Operating System settings from the Jukebox Server. When done this way, Jukebox Player will display a dialog warning the user. The dialog provides a warning that testing indicates that the application works with the specified Windows OS though the vendor information does not specify the OS as compatible:

- **Windows 8/8.1/2012/2012 R2 (64-bit)** – Check this box if the 64-bit versions of Window 8, Windows 8.1, Windows Server 2012, and/or Windows Server 2012 R2 are supported by the application since these Windows platforms have the same requirements.
- **Windows 7/2008 R2 (64-bit)** – Check this box if the 64-bit versions of Window 7 and/or Windows Server 2008 R2 are supported by the application. Windows 7 and Windows Server 2008 R2 have the same requirements.
- **Windows Vista (64-bit)** – Check this box if the 64-bit version of Windows Vista is supported by the application.
- **Windows 8/8.1 (32-bit)**– Check this box if the 32-bit versions of Windows 8 and/or Windows 8.1 are supported by the application. Windows 8 and Windows 8.1 have the same requirements.
- **Windows Vista/7/2008 (32-bit)** – Check this box if the 32-bit versions of Windows Vista, Window 7 and/or Windows 2008 Server are supported by the application. Windows Vista, 7, and Server 2008 have the same requirements.
- **Windows XP/2003 (32-bit)** – Check this box if the 32-bit versions of Windows XP and/or Windows Server 2003 are supported by the application. Windows XP and Server 2003 have the same requirements.
- **Compatibility Settings** – This button brings up a dialog that provides several compatibility options to be used when launching the cloudified application. For instance, this may be needed for applications designed for an operating system older than the target operating system(s). Which options are accessible depends on the selected target operating systems. For instance, if only Windows XP is selected, then new compatibility settings introduced in Windows Vista will be disabled. For more information, see About OS Compatibility Settings on page 40.

CAUTION

Appsets that are created on a 64-bit platform, regardless if the application is 32 or 64 bit, can only be modified or paged to a 64-bit target OS. Appsets created on a 32-bit OS can be used on a 64-bit OS, but may not always work depending on the application requirements.

Data Compression

Compression is used to improve overall cloudpaging performance experienced by the end-user. The greater the compression ratio applied to data pages, the smaller and faster an appset will page. There are three page compression settings to choose from:

- **None** – Indicates that no compression will be performed. This is useful for testing an appset, but should never be used for paging.
- **Fastest (LZSS)** – Uses LZSS compression technique (widely used in the industry) and is fast to compress and decompress. This is useful to reduce cloudifying times and satisfactory to page on a faster network.
- **Best (LZMA)** – (default) Uses LZMA compression techniques and is fast to decompress but takes longer to compress. This is the suggested compression method as it results in typically 20% smaller appset size than the LZSS method, which results in a greater transfer rate when paging.

NOTE

During the initial appset creation, optimization, and testing phases, it is most efficient to use “None” as the compression setting, in order to speed up the cloudifying process as much as possible. Then, after Optimization is completed, the compression can be re-set to the optimal level based on the particular application requirements and user needs.

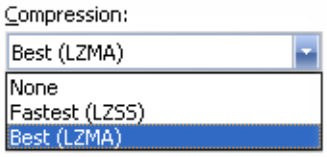


Figure 3-4: Data Compression drop-down menu

Data Encryption

Encryption is used to help protect the appset as it is delivered and stored by Jukebox Player. The better the encryption method is applied to data pages, the harder it is to steal the content stored in that page of data and the more secure the application will be. There are three page encryption methods choose from:

- **None** – Indicates that no encryption will be performed. This is useful for testing an appset but should never be used for real application delivery.
- **Fastest (Bitshift)** – Uses basic bitshifting technique that is not highly secure, but does not incur much performance overhead when decrypting. This is useful to reduce cloudifying times and satisfactory to page to slower PC's.
- **Best (AES 256)** – (default) Uses the Advanced Encryption Standard (AES) with a 256-bit symmetric-key and a Cipher Feedback mode to provide the best level of security for the data but takes longer to decrypt. This is the suggested encryption method as it results in appsets that are more secure than the other options.

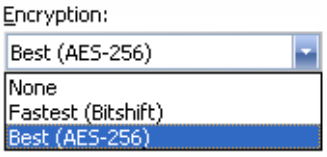


Figure 3-5: Data Encryption drop-down menu

Other Application Settings

Additional, optional settings are accessible via the Application Information dialog.

To change the Application Information Settings

From the Project view, click the **More Information...** button. The Application Information dialog appears.

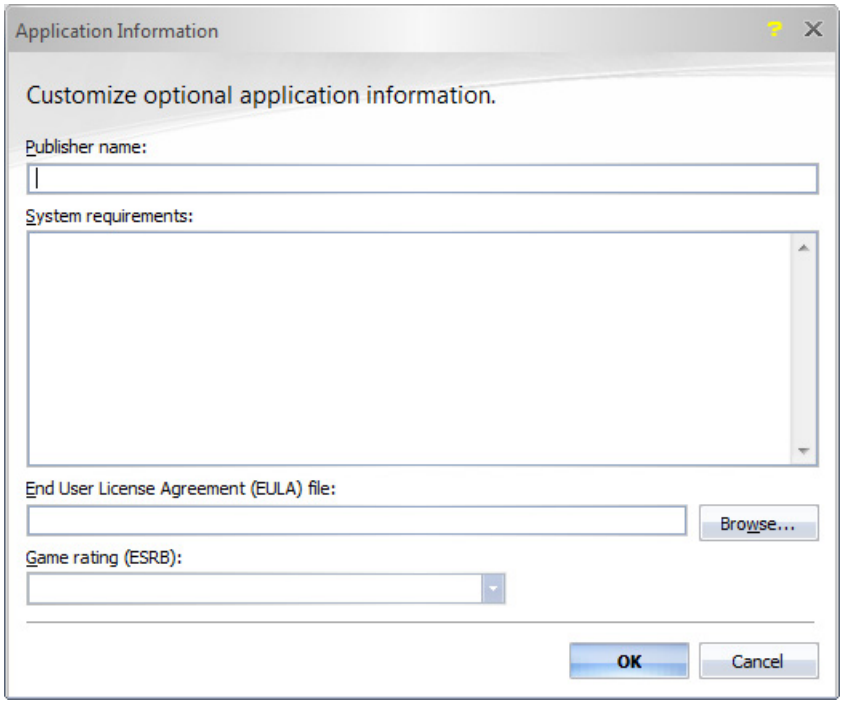


Figure 3-6: Application Information Dialog

- **Publisher Name** – The name of the company that published the application or game, such as Microsoft or Electronic Arts.
- **System Requirements** – These are the hardware and software requirements provided by the manufacturer and are typically printed on the software packaging. These requirements are for informational purposes only and can also be viewed from the Application Jukebox Server UI.
- **Application icon** – The target file for the icon is automatically populated when an .exe file is set as the launch command. An icon can be set manually using an .ico file or by extracting an icon from any .exe or .dll file.

NOTE

When extracting an icon from an .exe or .dll file, Jukebox Studio will use the first icon (Index 0) in that file.

- **Application End User License Agreement (EULA) file** – To populate this field, navigate to the EULA file. This is typically an HTML file that consists of a legal agreement the user acknowledges regarding the warranties and conditions of use for the software program.
- **Game Rating** – This drop-down menu provides options for parents to select age-appropriate games for their children. **None** is the default setting, and is typically used for business applications.

Step 3: Start the Application Installation Capture Process

Before you start the capture process, it is recommended that you save the project. You should also become aware of where files and registry settings are stored. The more you know about this before the capture process, the easier it will be to optimize the appset later.

Before launching the capture process, you can select the default disposition for new files, folders, and registry keys via the Global Disposition setting.

For more information on disposition, see Virtualization on page 7.

For more information on changing this setting, see Globally Changing Asset Dispositions on page 113.

To start the capture process:

1. Click **Capture** on the Jukebox Studio toolbar.

Figure 3-7: Capture button on the toolbar

Alternative: Click **File** on the menu bar, then select **Capture Application**.

The Capture Application Installation dialog appears.

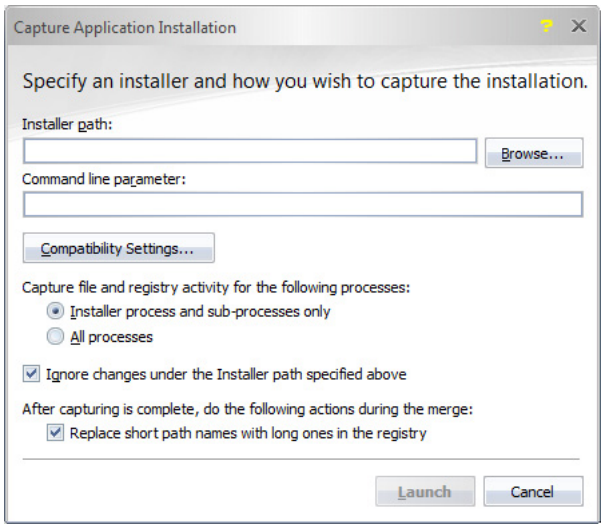

Capture Application Installation
Specify an installer and how you wish to capture the installation.
Installer path:
Browse...
Command line parameter:
Compatibility Settings...
Capture file and registry activity for the following processes:
Installer process and sub-processes only
All processes
Ignore changes under the Installer path specified above
After capturing is complete, do the following actions during the merge:
Replace short path names with long ones in the registry
Launch
Cancel

Figure 3-8: Capture Application Installation Dialog

2. Click **Browse** and navigate to the application's executable (**.exe**) or installer file (**.exe** or **.msi**) for the **Installer Path**. While this may typically be an installer, it could be the application's main executable depending on what needs to be captured.

NOTE

It is recommended to copy the installer under a temporary folder and launch it from there. This can help to avoid artifacts compared to launching it from a common location like the desktop.

3. If any arguments are required to run the installer, supply them on the **Command line parameter** field.

4. Under **Capture file and registry activity for**, select either **Installer process and sub-processes only** (default) or **All processes**. Then if required, select **Compatibility Settings**.
   - **Installer process and sub-processes only** – When this option is selected, Jukebox Studio will record only file and registry activity coming from the selected process and its sub-processes. All other system activity is ignored.
   - **All processes** – When this option is selected, file and registry activities coming from all processes on the system are recorded. Unwanted artifacts could result from capturing All Processes. For example, browsing the Internet while capturing the installation of an application may cause a change in the browser cache, which will also be captured.
   - **Compatibility Settings** – This dialog provides several compatibility options to be used when an installer is launched. For instance, enabling **Compatibility Mode** may be necessary for installers designed for an operating system older than the Cloudifying PC operating system.

     Which options are accessible depends on the Cloudifying PC operating system. For instance, under Windows XP, new compatibility settings introduced in Windows Vista will be disabled. For more information, see *About Compatibility Settings* at the end of this chapter.
   - **Ignore changes under Installer path** – This option can be used if you are capturing the application and not an installer. This option is selected by default to exclude temporary files created in this location by many installers.

CAUTION

If **Ignore changes under Installer path** is checked, installing an application to the same folder as the installer will cause files needed for proper capture to be excluded from the package. We recommend that the install source path and the install target path be separate folders.

   - **Replace short path names with long ones in the registry** – Some applications still refer to their component files by their 8.3 DOS-style file and folder names. For these applications, uncheck this box.

NOTE

Jukebox Studio will filter the assets and processes it captures based on the contents of the configuration DAT files detailed in **Appendix B**, Editing Configuration Files on page 167.

5. Click **Launch**.

   The Capture in Progress dialog appears and the application installer launches.

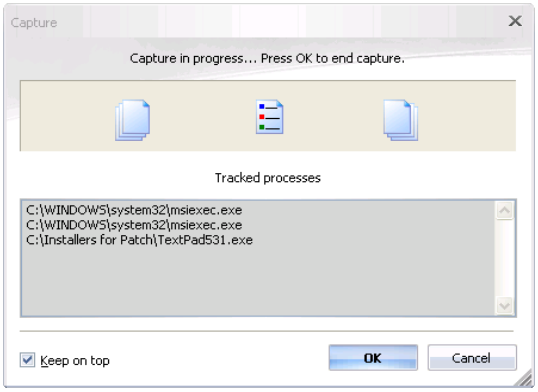


Figure 3-9: Capture In Progress dialog

During the capture, the bottom half of the Capture in Progress dialog shows a list of every process captured during the installation.

NOTE

Leave this dialog open. Later, when the application installation process has completed and you want to end the capturing process, you will click **OK**.

Step 4: Install the Application

Follow the application installer instructions to complete the application installation. Be sure you install all components during the installation process to minimize any dependencies. Install the application to the Install Folder.

Most applications have Standard and Custom installation options. It is recommended that you use the **Custom** option so that all application components will be available to users of varying needs.

Install the application on the same hard drive on the Cloudifying PC that contains the Windows directory. This is where the target application must be installed.

Installation Folder Restrictions

The folder where the application will be installed should not be the root directory of the hard drive. This will cause Jukebox Studio to filter nothing. Though technically this will not prevent proper application paging, the appset will contain unneeded data and be larger than required. Do not use special characters in the installation directory name (e.g. no umlauts, ~, !, @,#, à. Á etc.).

CAUTION

The application has to be installed to a folder on the same drive as Windows. Other disk drives will not be captured.

We recommend that the installed application should not be in the same folder as the installer executable or in a sub-folder of the installer folder.

We also recommend that you install to the default location, such as Program Files, but with a unique folder name. Be sure this folder name is unique as it will help to avoid issues with legacy application that still use the older 8.3 naming convention (short file names) that can cause integration conflicts.

Avoiding Artifacts

Artifacts are unwanted changes or additions that occur most often when “All Processes” is selected on the Capture Application Installation dialog. Follow the guidelines below to avoid or reduce artifacts.

- Avoid running applications or executing commands that are not related to the target application on the cloudifying machine while capturing installation and other functions are taking place.
- Do a dry run of the installation on a machine other than the one used for cloudifying so that you know what to expect.
- In general, minimize all machine activity after any cloudifying activities begin.

CAUTION

It is recommended to turn off any power saving features that would cause the system to go standby or hibernation during the capture process.

If you selected the “launch program after installation” option at the end of the application installation, continue to Step 6. If not, skip to Step 7.

Step 5: Run the application (optional)

The purpose for running the application at this point is to capture settings (such as registry modifications) that are made after the installation is completed. Such cases are processes such as displaying the EULA (End User License Agreement) or changing user profile settings. After the installation is complete, it is advised to run the application and try using some menu options and features to verify that related components are properly installed. Some suggested actions are:

- Start the application
- Click and use menu items
- Click and use Toolbar items
- Perform a typical task and usage scenario for the application. Refer to the user guide for the application.
- Close the application

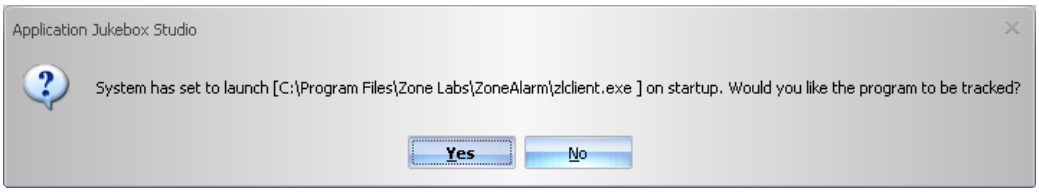
NOTE

For applications that require running other processes after the installation is complete, you may need to stop the capture, then start it again for each process.

Step 6: Capturing System Reboots

If the installation requires a system reboot after the installation is complete, then take the following steps:

1. Click the **restart** button from the application’s installer. Do not end capture or exit Jukebox Studio, as it will save work in progress and shutdown on it’s own.
2. Once the system restarts, login is required.
3. Jukebox Studio will startup automatically in capturing mode and will prompt you if there are additional Run key process that can be captured.


Application Jukebox Studio
System has set to launch [C:\Program Files\Zone Labs\ZoneAlarm\zlclient.exe] on startup. Would you like the program to be tracked?
Yes
No

4. Select **Yes** to the additional processes that you wish to capture. You will be prompted for each run key contained in the registry.

NOTE

There may be other process set to run that are not part of the application installation. Please be sure to select the correct processes.

Step 7: Stop the Installation Capture

When the installation is complete, the Capture in Progress dialog will display a message that the capture is complete.

To end the capture process, click **OK.**

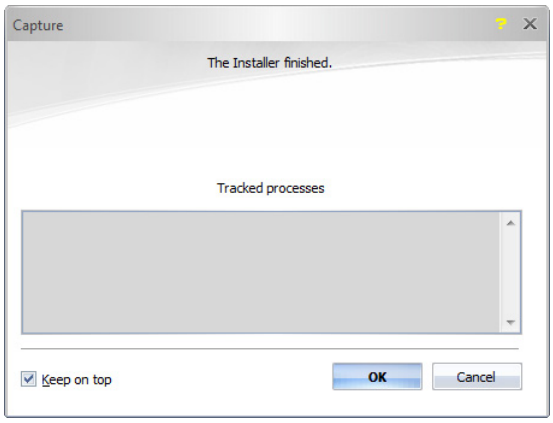


Figure 3-10: Capture in Progress dialog, completed

Be sure that the installation is complete before you stop the capture process. After you stop the capture process, the captured assets will be added (merged) into the project, and the project workspace will be updated to reflected these changes.

The Merge dialog will show which files and registry keys were filtered, merged, or had their security descriptions changed during the merge.

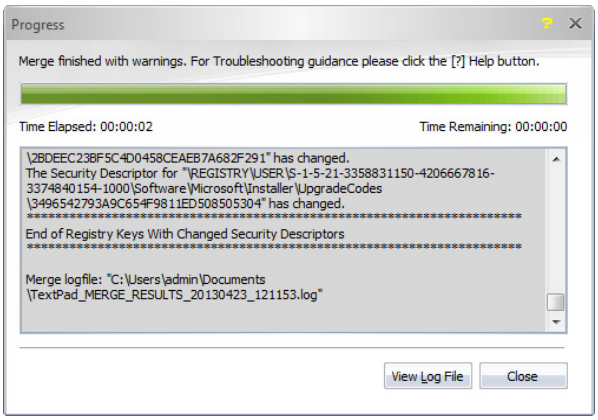


Figure 3-11: Merge dialog

This information might be useful for troubleshooting. For example, if the installer changes the security settings for a file, you might need a Configurable AppEvent (CAE) to perform the same task on the Target PC.

Jukebox Studio also saves the merge information to the log file. To see this, click **View Log File**.

NOTE

Occasionally, you may see a message about non-critical errors during the merge of the captured data. These are due to protected objects by the Operating System not being accessed and can safely be ignored.

If you see an error message about security descriptors, User Accounts, or Plug-n-play drivers, you will need to create a Configurable AppEvent to handle these. See Configurable AppEvents Settings on page 58 for more information.

There is no limit to the number of captures that you make. At any time you can re-configure project settings and take another capture.

Step 8: Enter Remaining Application Information

After capturing the installation application, there are a few more pieces of information that Jukebox Studio will require before the next step.

Revision Information

The Revision Information sets the name of the current patch version, command line, and working folder for the application.

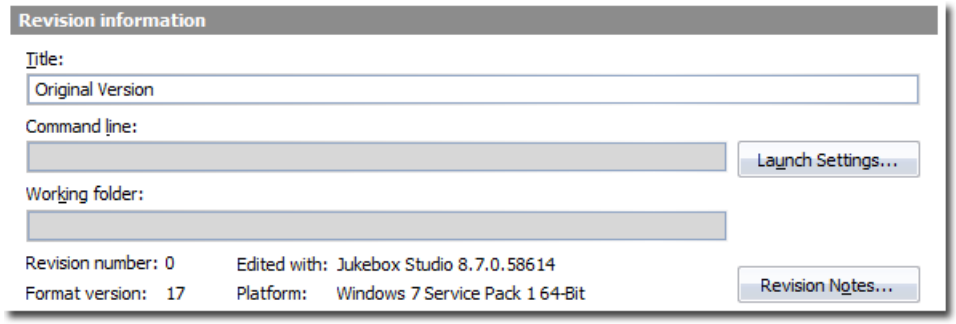


Figure 3-12: Revision Information

- **Title** – This is the patch name of the current version of the appset. It defaults to “Original Version” the first time you capture an application. When you create a patch (**Chapter 10**, Patching an Appset on page 157), you can assign the patch a description name, such as the actual version number and/or the release date.
- **Command line** – Specifies the command line used to start the program. When the user clicks the **Launch** button on Jukebox Player, the executable in the Command Line executes. The easiest way to populate it properly is to navigate in the Files view to the application’s executable that is to be launched, right-click it, then select **Set as command line**. Once set, the file marked as the Command Line will appear in **bold** in the Files view.

  If you’re not sure where the executable is located, one way it can be found is by looking at the properties of the shortcut created when the application was installed.

CAUTION

Setting the **command line** manually from the Project View will not set any security defaults for the file chosen. In addition, file paths containing templates (see About Folders on page 73) will need to be substituted manually.

- **Working folder** – Specifies the folder that contains the application's executable or some related files. A vast majority of the time, using the same directory as used in the command line will work. On occasion using the same directory as where the program executable is found will not work. In these cases, it may be required to set the Working Directory to the application root directory. The easiest way to populate this field is to navigate in the Files view to the folder that you want to set as the working folder, right-click it, then select **Set As Working Folder**. The working folder path will be shown in **bold** in the Files view.
- **Revision Number** – If this appset has been patched, this number will show the number of patches that have been applied.
- **Format Version** — This number indicates which version of the Studio packager was used to create the current Appset. This information can be useful if you need to contact Numecent technical support.
- **Platform** — This indicates which version of the operating system was in use when the appset was created.
- **Notes** — Click this button to add or read any notes relating to this appset.

To Set the Command Line and Working Folder from the Project View

1. Click the **Launch Settings...** button. The **Application Launch Settings** dialog appears.

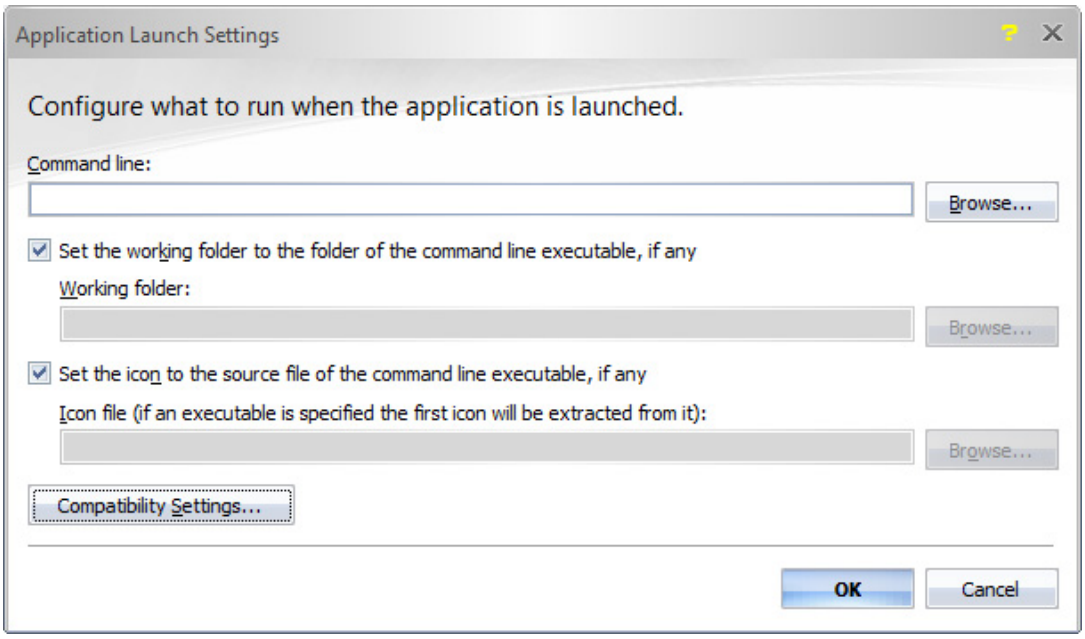
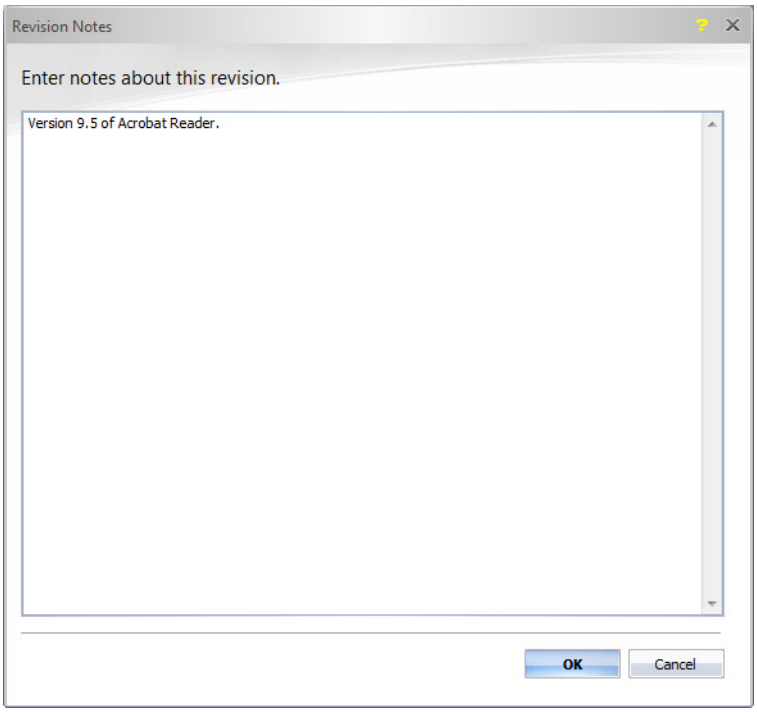

Application Launch Settings
Configure what to run when the application is launched.
Command line:
Browse...
Set the working folder to the folder of the command line executable, if any
Working folder:
Browse...
Set the icon to the source file of the command line executable, if any
Icon file (if an executable is specified the first icon will be extracted from it):
Browse...
Compatibility Settings...
OK
Cancel

Figure 3-13: Application Launch Settings

2. To set the **Command Line**, type the path to the desired executable file or browse for it.
3. To set the **Working Folder**, type the path to the desired folder or browse for it.

Revision Notes

Revision Notes provide you with a space to enter information regarding each revision of the appset. To enter notes for the initial version, click the **Revision Notes...** button. The **Revision Notes** dialog appears.



A new revision note is entered every time the appset is patched. Each revision note is saved as a separate text file in the appset file.

For information on patching, see **Chapter 10**, Patching an Appset on page 157.

Step 9: Save the Project

Click **Save** on the Jukebox Studio toolbar to save the project (**.stw**) file.

About OS Compatibility Settings

The **Compatibility Settings** dialog offers several options that affect the environment in which a program is launched. Those options are useful for programs designed to run on older operating systems than the ones on which they will actually be run. For Jukebox Studio, this dialog appears under two different conditions: the program may be the installer for the application being cloudified or the cloudified application itself.

If you change compatibility settings for a given program, then Windows automatically modifies the run-time environment upon launching the program to make it as close as possible to the environment required for the program to function properly.

Compatibility Settings options

Certain compatibility settings apply only to some of the supported operating systems. This is clearly indicated next to each setting.

NOTE

Since multiple target operating systems may be selected, Jukebox Studio allows you to choose any of the options, even those that do not apply to some of the target operating systems you selected. However, any setting that does not apply to a particular OS will be ignored when the application is paged on that OS.

- **Run this program in Compatibility Mode** – This setting is used when a program is only compatible with an operating system older than the one on which it will run. In particular, since Jukebox Studio only runs on Windows XP or later, this option should be used for applications that only run properly on prior operating systems such as Windows 9x or Windows 2000. If the application indicates compatibility with Windows XP or later, then this option does not need to be selected. Otherwise, as a rule of thumb, the most recent supported operating system should be selected for compatibility mode.
- **Reduced color mode** – This setting forces a program to run with a limited set of colors. Two options are available: **8-bit (256) color** and **16-bit (65536) color** (Windows 8 and later).This may be necessary for older programs, especially games, that do not support higher color ranges such as 24-bit or 32-bit color.
- **Run in 640x480 screen resolution** – This setting forces the entire screen to 640x480 resolution while the program is running. This may be necessary for older programs, especially games, that do not support higher resolutions. Try this setting if the graphical user interface appears jagged or is rendered improperly.

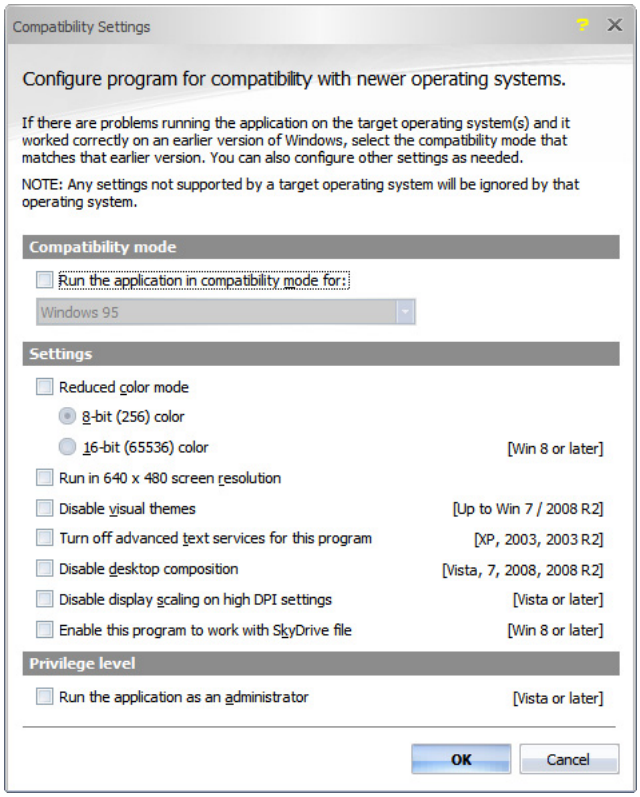

Compatibility Settings
Configure program for compatibility with newer operating systems.
If there are problems running the application on the target operating system(s) and it worked correctly on an earlier version of Windows, select the compatibility mode that matches that earlier version. You can also configure other settings as needed.
NOTE: Any settings not supported by a target operating system will be ignored by that operating system.
Compatibility mode
Run the application in compatibility mode for:
Windows 95
Settings
Reduced color mode
8-bit (256) color
16-bit (65536) color
[Win 8 or later]
Run in 640 x 480 screen resolution
Disable visual themes
[Up to Win 7 / 2008 R2]
Turn off advanced text services for this program
[XP, 2003, 2003 R2]
Disable desktop composition
[Vista, 7, 2008, 2008 R2]
Disable display scaling on high DPI settings
[Vista or later]
Enable this program to work with SkyDrive file
[Win 8 or later]
Privilege level
Run the application as an administrator
[Vista or later]
OK
Cancel

Figure 3-14: Compatibility Settings Dialog

- **Disable visual themes** (Windows 7 and earlier) – This setting turns off all visual themes while the program is running. Try this setting if you notice problems with the menus or buttons on the title bar of the program.
- **Turn off advanced text services for this program** (Windows XP only) – This setting disables advanced text services such as speech recognition.

- **Disable desktop composition** (Windows Vista/7 only) – This setting turns off transparency and other advanced display features. Choose this setting if window movement appears erratic or you notice other display problems.
- **Disable display scaling on high DPI settings** (Windows Vista and later) – This setting turns off automatic resizing of programs if large-scale font size is in use. Try this setting if large-scale fonts are interfering with the appearance of the program.
- **Enable this program to work with SkyDrive files** (Windows 8/8.1 only) – This setting allows a program to see and open files on SkyDrive that may not be otherwise visible.
- **Run this program as an administrator** (Windows Vista or later) – This setting causes Windows to elevate the program to administrator level before running it. Some programs require administrator privileges to run properly.

For more information on a particular setting, search one of the following web sites:

- **http://support.microsoft.com/**
- **http://windowshelp.microsoft.com/**

Chapter 4

Cloudifying the Application

This chapter covers these topics:

- **Overview**
- **Procedure for Cloudifying the Application**
  1. Configure Cloudpaging Settings
  2. Modify Files, Registry, and Other Assets
  3. Cloudify the Application

Overview

As described in *Cloudifying Overview* chapter, application cloudifying with Jukebox Studio has four phases. **Phase 3: Cloudify the Application**includes the steps necessary for creating an *appset*, or "cloudified application." In addition, procedures for modifying the content and settings of files, registry, and environment are provided. These modification procedures also apply to Phase 5 (testing and optimizing the appset), as well as modifying a patched appset (Chapter 5).

Figure 4-1: Phase 3 of the Cloudifying Process

Procedure for Cloudifying the Application

Step 1: Configure Cloudpaging Settings

Check the Settings dialog and modify any of the setting which affect the behavior of the cloudified app on the Target PC.

- **Paging Settings** ..... page 56

  This configures additional paging sources to improve performance of the cloudified application.
- **Configurable AppEvents Settings** ..... page 58

  This configures the cloudified application to launch scripts at different stages of the paging process to perform specific actions on the Target PC as required by the application.

- **Virtualization Settings** ..... page 62

  This controls how the cloudified application interacts with the Target PC.
- **Sandboxing Settings** ..... page 63

  The cloudified is kept isolated from other programs and assets on the Target PC, but this can be modified or disabled for specific assets using these setting.
- **Security Settings** ..... page 66

  This allows you override file security settings for a specific asset if required for the cloudified application to work on the Target PC.
- **Performance Settings** ..... page 69

  This setting controls how much of the cloudified application can be stored locally by the Jukebox Player software.

NOTE

Before cloudifying, you should check the compression and encryption settings on the Project view to make sure those are set to the desired options. See Data Compression on page 30 and Data Encryption on page 31.

Step 2: Modify Files, Registry, and Other Assets

Noticing differences in Files, Registry and other assets before and after the capture process will provide a basis for what needs to be modified.

CAUTION

Before cloudifying, you should check these view to see if there are any conflicts (items in red) and determine if these represent actual problems that could affect cloudifying or cloudpaging.

- **Modifying Files and Folders** ..... page 72

  Add or remove files and folders from the project, or change their disposition.
- **Modifying Registry Entries** ..... page 83

  Add or remove registry keys and values, and change existing values.
- **Modifying Environment Variables** ..... page 89

  Add or remove environment variables.
- **Modifying Fonts** ..... page 92

  Add or remove fonts.
- **Modifying Startup Items** ..... page 96

  Add or remove startup items, including registry keys and Startup Menu items.
- **Modifying Services and Drivers** ..... page 102

  Add or remove Windows services and software drivers.

Step 3: Cloudify the Application

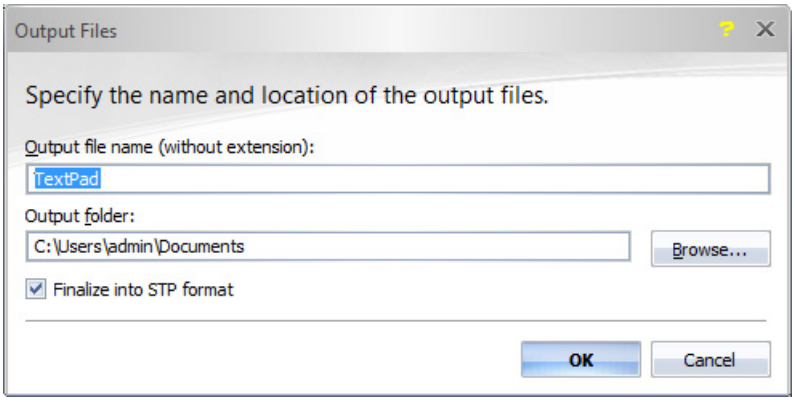
After you provide all information and configuration settings in the Project, File, Registry, and Environment views, you are ready to create the appset.

To cloudify the target application:

1. Click **Cloudify** on the Jukebox Studio toolbar.

Figure 4-2: Cloudify button on the toolbar

2. The **Output Files** dialog appears:



This dialog lets specify a name and an output folder for the final appset.

By default, **Finalize into STP format** will be checked. Jukebox Studio will compress the components into the containing STP file. If the box is unchecked, the resulting STC file (the code and other configuration information that comprises most of the application), AIB file (metadata that describes the content of the STC file), TOK file (comprised of security data), and any other files such as icons, AppEvents, and EULA's, will be saved to the output folder.

NOTE

For instructions on how to manually finalize the cloudified app into the STP format, see Saving an Appset File on page 144.

Saving the components without compressing them is much faster than creating the STP file. This allows you to check and edit the resulting AIB and TOK files, before creating the finished Appset.

CAUTION

Periods are not allowed in the appset filename. If you manually rename the appset, be sure NOT to use more than one period in the filename. Using multiple periods will prevent Jukebox Server from adding the appset.

NOTE

Some software, such as virus scanners, can have a negative impact on the performance of Jukebox Studio during the cloudifying process. You may wish to disable such software before you cloudify the application.

When the appset is later *published* to the Jukebox Server, the target application (or appset) becomes available for paging to users.

Chapter 5

Testing and Optimizing the Appset

This chapter covers these topics:

- **Overview**
- **Testing the Appset**
  1. Test the appset
  2. Identify and Solve Problems
- **Optimizing the Appset**
  1. Create Prefetch Files
  2. Set Compression
  3. Set Encryption
  4. Adjust Cache Size
  5. Cloudify the Application
  6. Optional: Save the Cloudifying Environment
  7. Publish the Appset to Jukebox Server

Overview

As described in the *Cloudifying Overview* chapter, application cloudifying with Jukebox Studio has four phases. Phase 4: Testing and Optimizing the Appset includes the steps necessary for testing, optimizing, and re-testing the appset each time a setting changes. When the appset reaches optimal performance, the cloudifying process is complete and the appset can be published to Jukebox Server.

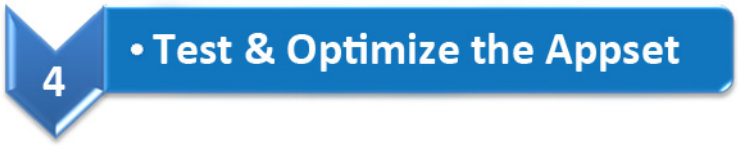


Figure 5-1: Phase 4 of the Cloudifying Process

Testing the Appset

Step 1: Test the appset

To test the appset, you will load it onto a Target PC with Jukebox Player installed, then run the application. *Local paging*, the process of paging an appset to Jukebox Player from a file located the Target PC, allows you to test several appset iterations quickly without publishing the appset to Jukebox Server.

It is necessary to go through as many testing scenarios as are required, for different operating systems, application versions, etc. In this case, a clean version of the Cloudifying PC needs to be made and a new appset created for each different installation.

The Target PC can be one machine or multiple machines. Multiple PC's would allow testing to occur in parallel. The only requirement for the Target PC is that a clean installation of the operating system be present.

CAUTION

Be sure not to use the Cloudifying PC also as the Target PC to test application paging. This is mainly because at the end of the installation, the application resides entirely on the Cloudifying PC. So if the paged version of the application is tested on the Target PC and a component is missing from the content (STC) file, Jukebox Player is likely to find the installed component instead, which would give the appearance that the application is working fine.

To test the appset:

1. Place [AppsetName]**.stp** on the Target PC with Jukebox Player installed.
2. Rename the **.stp** file to [AppsetName]**.zip**, and then right-click it and select **Extract Here**.
3. Double-click the **.tok** file. It will activate (outside of Jukebox Player).
4. Click **Launch** on the Jukebox Player toolbar. The target application will run.
5. At this point you can test the appset.

For example, all menu options should be tested to verify that all parts of the application are installed and captured. If the application does not work as expected, then some part of the target application installation was not completed properly.

Verify that the default location where the application saves files to is already part of the Excluded Paths for sandboxing rules (see Sandboxing Settings on page 63). This will ensure that user files remain on the system if the virtual application is removed.

NOTE

Optimizing an appset is typically an iterative process that requires testing between each successive cloudifying trial, as well as after each change in files, registry, or environment.

Step 2: Identify and Solve Problems

If you encounter problems or unwanted *artifacts* (dlls, etc.) when you test the appset, follow this procedure for identifying and solving problems:

- **First, identify the problems.**
- **Second, modify the appset's files, registry, or environment to correct the problems.**

Optimizing the Appset

After you have corrected existing problems, it is best to tune the performance of the appset. Delivery of applications can be optimized in two ways: by optimizing paging and/or by caching. To assist with these optimizations, please follow these basic steps.

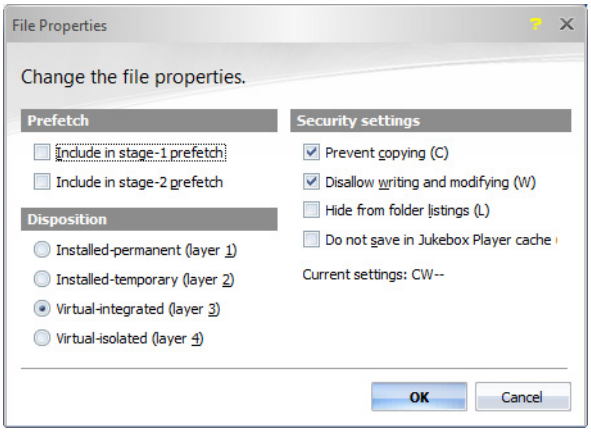
Step 1: Create Prefetch Files

One of the best ways to optimize the performance of the appset is to create a Stage-1 prefetch, and possibly a Stage-2 prefetch file. *Prefetching* is the process of fetching application pages prior to their being explicitly requested by Jukebox Player, in anticipation of being needed by the application. For more information, see Prefetching on page 7.

- **Stage-1 Prefetch** – Application pages needed to start the application. Always recommended for best performance to launch an application.
- **Stage-2 Prefetch** – Progressive background fetching of additional application pages after application starts, improving overall performance.

To manually add a file to the prefetch:

1. View the **Files** tab of the Jukebox Studio workspace.
2. Select the file to be added to the prefetch and right-click to select **File Properties**.
3. Check the **Prefetch** box for adding to either **Include in stage-1 prefetch** or **Include in stage-2 prefetch**.
4. Click **OK**.



To capture and add a prefetch:

1. Enable the prefetch **Capture** button on Jukebox Player: right-click the Jukebox Player shortcut on the Windows deskop, and select **Properties**.

   The **Jukebox Player Shortcut Properties** dialog appears. See Figure 4-2.

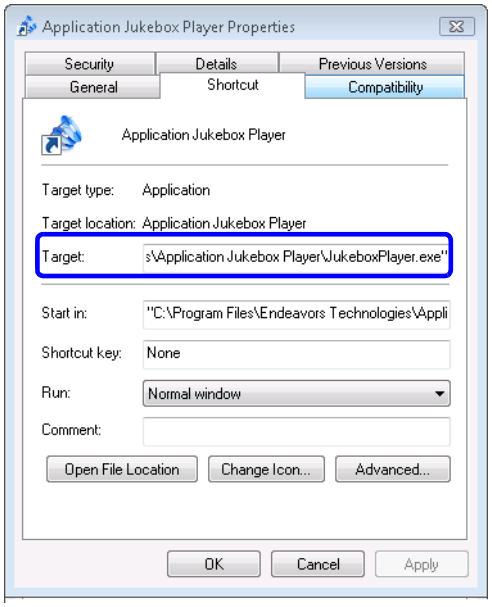


Figure 5-2: Jukebox Player Shortcut Properties Dialog

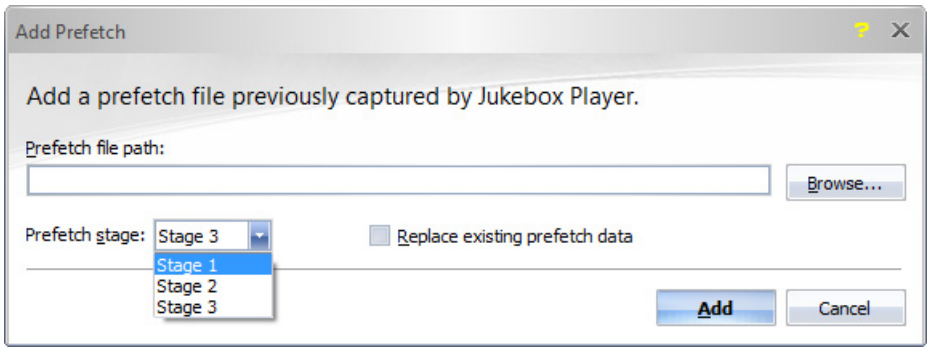
2. On Target, append **/CapturePrefetch** to the path as follows: **"C:\Program Files\Numecent\Application Jukebox Player\JukeboxPlayer.exe" /CapturePrefetch**.
3. Click **OK**, and launch Jukebox Player.

   Jukebox Player launches with the Prefetch **Capture** button enabled.

Figure 5-3: Jukebox Player toolbar with Prefetch Capture button enabled

4. Add the appset to the Jukebox Player by paging it either locally or from a Jukebox Server.
5. Clear the Player cache. (Click **File**, select **Options**, click **Clear Cache**, click **OK**.)
6. To start prefetch capturing, select the **Stage** and click **Capture** on the Jukebox Player toolbar.
7. A save dialog will appear for the filename to store the captured information. By default this will save to **My Documents** as <application's short name>_<application ID>_Stage_<stage #>.pf.
8. From the target application, perform the actions to include in this stage of prefetch, then click **Capture** to end the capture process.
9. Click the **Capture** button on the Jukebox Player toolbar to stop capturing.
10. If you wish to add another stage prefetch, repeat steps 4 - 9 by selecting different **Stage** numbers.
11. Open the Jukebox Studio project file for that application.
12. From the toolbar, click the **Prefetch** button.

13. The Add Prefetch dialog appears.



14. Enter the prefetch file that was just captured by Jukebox Player into the **Prefetch file path** field.
15. Select the **Prefetch stage** for which the prefetch file will be used.

NOTE

If one or two prefetch stages are in use, three prefetch stages will be shown in the dropdown list. Applying the new prefetch data to stage three will create a third preset stage. The next time the Add Prefetch dialog appears, four preset stages will be shown. For more information on how Jukebox Studio uses prefetch stages, see Prefetching on page 7.

16. Check **Replace existing prefetch data** if this will replace an older prefetch file previously added.
17. Click **Add** and repeat the process for other stages.

NOTE

If there is an existing prefetch stage, then the “replace existing prefetch data” check box will be enabled. This does not apply to files manually added to the prefetch.

Best Practices for Capturing Prefetch Files

When capturing prefect files, you should repeat the same activities you want to capture. For example, if you want to capture the launching part of an application into prefetch:

1. Turn on capturing.
2. Launch the application.
3. Stop the application, and launch it again.
4. Repeat while watching Jukebox tray icon.
5. Stop capturing when you no longer see Jukebox tray icon spinning as you launch the application.

NOTE

Many applications do not always read the same set of blocks every time you launch. An application will capture many of the same blocks during every launched but there are always some extra blocks that the applications will need at each launch, depending on circumstances. It normally takes 3 to 4 launches for an application (especially large applications) to capture most of the blocks usually requested during launch.

While capturing prefetch files on the Player:

1. Allow all prefetching to complete.
2. Stop other applications that might be running.

3. Clear the Player cache.
4. Start a prefetch capture for the application.
5. Launch the application.

Step 2: Set Compression

During the testing phase, compression is usually set at “None” or “Fastest” in order to speed up the cloudifying process. After testing is complete, you can change the compression level to “Best”, which is the recommended setting as it typically yields the best compression ratio. This will result in faster download times when paging the application. Be prepared that this can take some time to cloudify the application.

To adjust the compression:

1. Open the Jukebox Studio project file for the application.
2. Select the appropriate compression level under **Data compression**.

Step 3: Set Encryption

Typically the encryption method will be set to “Best” to ensure the highest level of security for the application content. However, encryption will slightly degrade application launch time performance due to the time it takes to decompress each page of data. If this overhead is not desired and security is **not** a concern, then the encryption method can be reduced to “Fastest” or “None” for improved application performance.

To adjust the encryption:

1. Open the Jukebox Studio project file for the application.
2. Select the appropriate encryption method level under **Data encryption**.

Step 4: Adjust Cache Size

Depending on the application, it may also be necessary to increase the allocated space in cache. A larger amount of cache space can improve application performance by temporarily storing more pages. Most application only need space for 5-10% of the total application in the cache. However, some application may need to cache more data at any given time for better performance. This setting will increase the minimum cache size required by the percentage of the uncompressed appset size as specified, plus the size of the prefetch stages (total maximum is 100%).

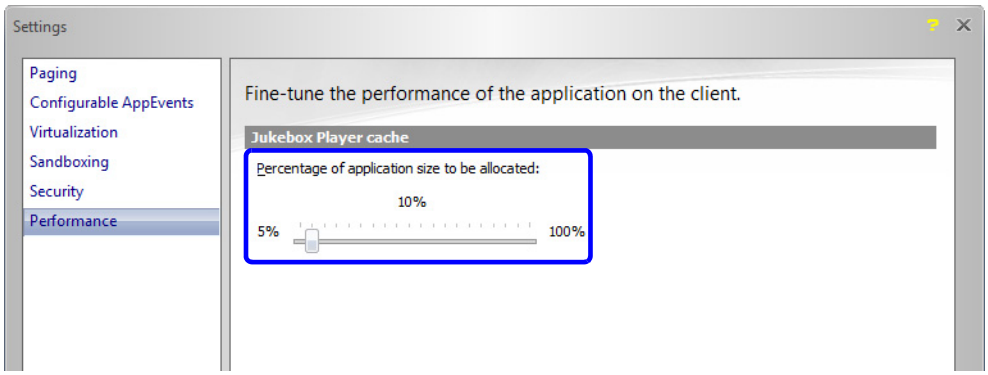
CAUTION

The user may be prompted to increase the Jukebox Player cache size. The cache size is limited by the available hard disk space on the user’s PC. If the user does not have enough cache space, then the application will not be allowed to activate.

To adjust the cache size:

1. Open the Jukebox Studio project file for the application.
2. From the Performance screen of the **Settings** dialog, adjust the **Percentage of application size to be allocated** under **Jukebox Player cache**.

3. Click **OK**.



Step 5: Cloudify the Application

After optimizing the application, you need to re-cloudify the appset.

To cloudify the application:

1. Click **Save** on the Jukebox Studio toolbar to save the project file (**.stw**).
2. Click **Cloudify** on the Jukebox Studio toolbar.

Step 6: Optional: Save the Cloudifying Environment

Patches can be created on any PC, as long as the most recent version of the appset has been extracted on that PC. If you wish to use the original Cloudifying PC for patching, follow these steps:

Save the project file (**.stw**), appset file (**.stp**), and any prefetch capture files (**.pf**). Or back up the cloudifying environment.

CAUTION

The original application must be installed in order for a patch to be properly integrated. This must either be done on the original Cloudifying PC or the appset can be extracted to another PC. The original project file (**.stw**) is also required.

Step 7: Publish the Appset to Jukebox Server

Once optimized, the appset is ready to be *published* from Jukebox Studio to the Jukebox Server, which is done by copying the final STP file onto the Jukebox Server repository. The application is then ready to be paged to users' PCs that are running Jukebox Player.

Chapter 6

Cloudpaging Settings

This chapter covers these topics:

- **Overview**
- **Paging Settings**
- **Configurable AppEvents Settings**
- **Virtualization Settings**
- **Sandboxing Settings**
- **Security Settings**
- **Performance Settings**

Overview

Cloudpaging settings control the behavior of the cloudified application when it is paged on the Target PC.

They are accessed via the **Settings** button or the **Settings** menu, which displays the **Settings** dialog.

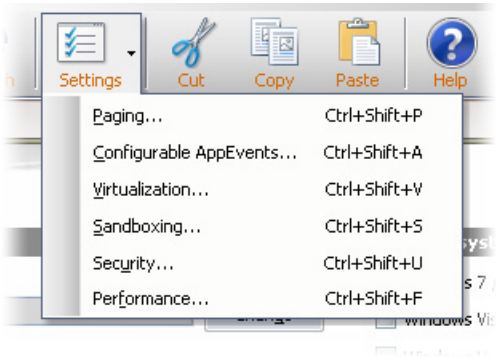


Figure 6-1: The Settings button with menu

To access the Settings dialog, either click the Settings menu and select Show Settings, or click the Settings button on the toolbar.

This dialog has several separate views, which are described in the corresponding sections below: **Paging**, **Configurable AppEvents**, **Virtualization**, **Sandboxing**, **Security**, and **Performance**.

To access a specific screen on the Settings dialog, select that screen name from the Settings menu. You can also click the arrow next to the settings button and select the screen name from the drop down menu.

The following sections discuss each settings category in detail.

Paging Settings

To improve performance, it may be desired to have data page located in a location closer to the Target PC. Paging sources make it possible to have multiple locations for the data page beyond that of a Jukebox Server. The data source can be a CD/DVD device or hard drive on the user's PC, or a company network drive (remote drive). Sources are paged from in sequential order of CD/DVD, Local/Remote Drive, and Jukebox Server. If data pages cannot be found at the first source, then the next source in the order will be attempted. The Jukebox Server will always contain all pages without exception.

To set up an additional data source:

1. Select **Paging...** from the Settings menu.

   The Settings dialog appears with the **Paging** screen visible.
2. Select **CD/DVD** or **Local/Remote Drive**, and then click **Enable Source**.
3. If **Local/Remote Drive** is selected, enter a file path or click Browse to locate the file path for this source.
4. Then select one of three options:
   - **Include only the following prefetch page** – Indicates that the selected prefetch blocks will be made available at the source location and add them to a new source STC file.
   - **Include all blocks** – Indicates that all data blocks will be made available at the source location and add them to a new source STC file.
   - **Strip random blocks** – Indicates that except for a random few, all data blocks will be made available at the source location and add them to a new source STC file. These random blocks are marked with

the security setting “Do not cache on local machine.” This is done for added protection as they must always be paged from another source.

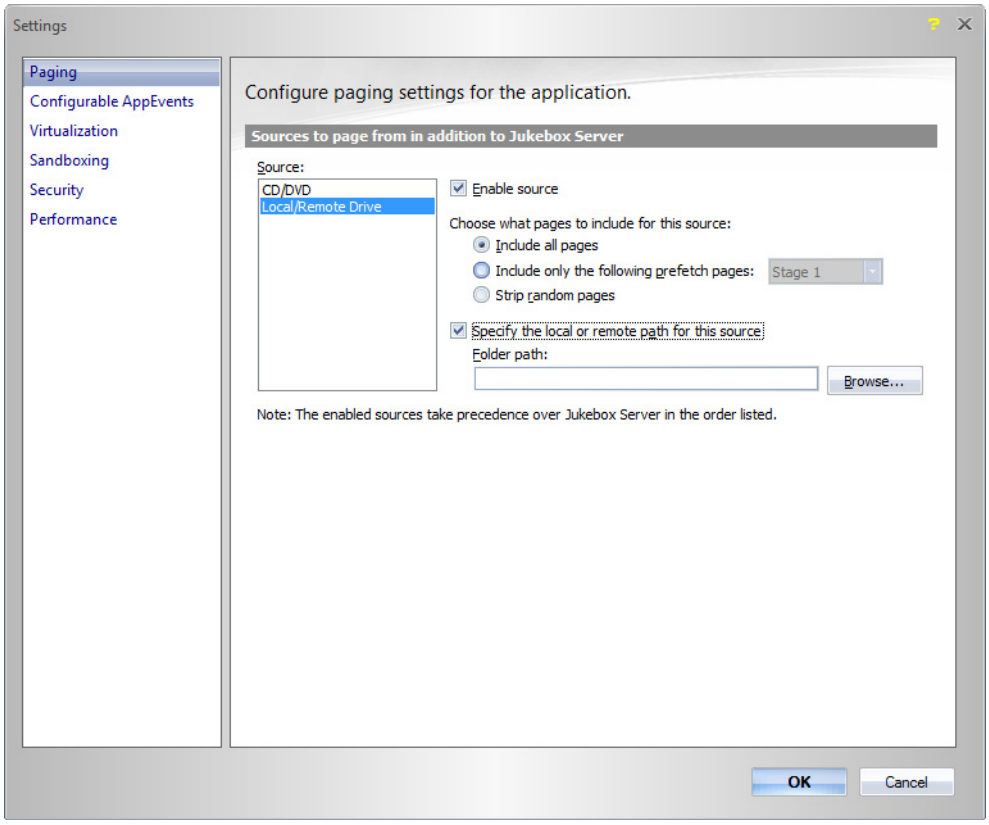


Figure 6-2: Settings Dialog, Paging screen

5. Click **OK**.

   When selecting a paging source, generating the appset will output an STP file and an additional STC file specific to the selected source (for example, **Appset_Name-CD.STC**).

NOTE

It is the responsibility of the ASP administrator to place the paging source STC in the appropriate location, on a CD or at the specified location.

To Create a partial STC

A partial STC can be used to distribute key parts of a cloudified application via a cloud storage service such as Dropbox, or any other shared resource, so that the local copy can be updated for all users at once.

1. Select **Paging...** from the Settings menu.

   The Settings dialog appears with the **Paging** screen visible.
2. Select **Local/Remote Drive**, and then click **Enable Source**.
3. Then select one of three the options. See above for details.
4. Uncheck **Specify the local or remote path for this source**.

Configurable AppEvents Settings

To convert a locally installed application into a cloudified application, Jukebox Studio captures the “native” states of the installed application and records them in the appset file. When an application is activated by the user, those states are re-created on the user’s system. Sometimes, to allow the application to run properly, it may be necessary to adjust those native states or the runtime environment before, during, or after activation. In other words, it may be necessary to perform certain actions that cannot be captured in an appset.

An *AppEvent* is an action that uses a *trigger* to run a *handler* (i.e. an executable program or script) that will perform specified runtime action(s) that are required by the application being cloudified, such as checking for prerequisites. AppEvent handlers can be written in any scripting or programming language, such as batch, vbscript, or C++ executables.

Examples of actions that may need to be performed at runtime are:

- Configuring plug-ins for office suites and browser applications
- Configuring virtual printers
- Configuring game parameters for a video game
- Starting a background program

To add an AppEvent

1. Select **Configurable AppEvent** from the Settings menu. The Settings dialog appears with the **Configurable AppEvents** screen visible.

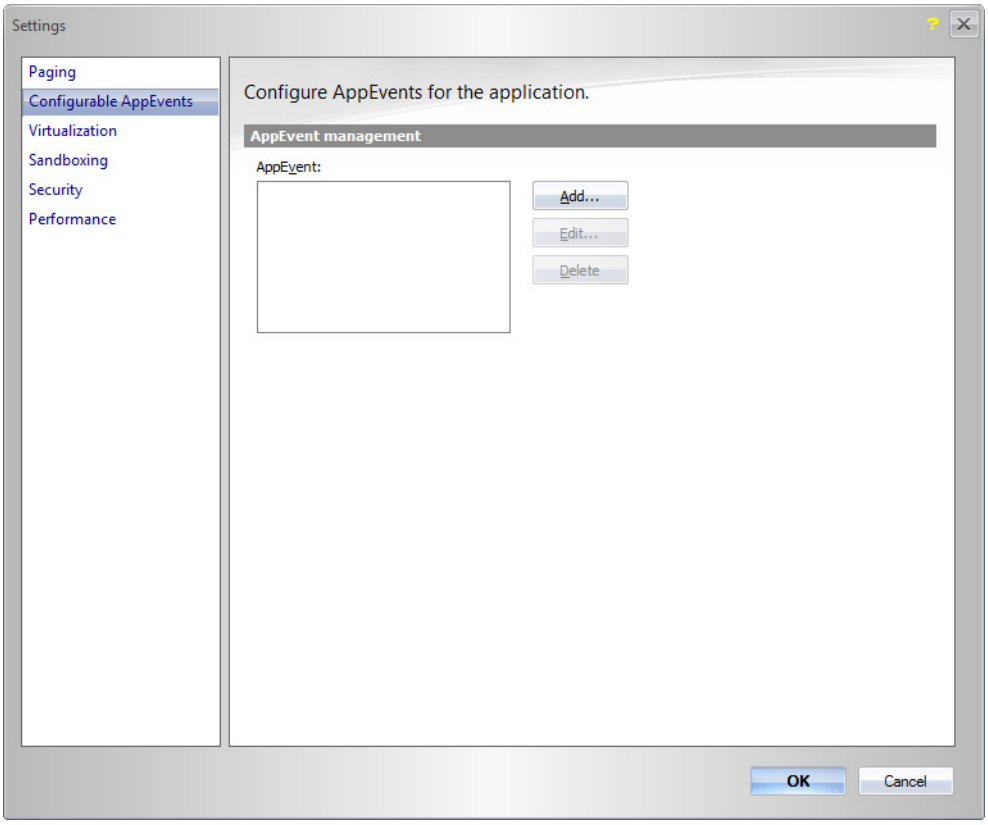


Figure 6-3: Settings dialog, Configurable AppEvents screen.

2. Click the **Add...** button. The **Configurable AppEvent Settings** dialog appears

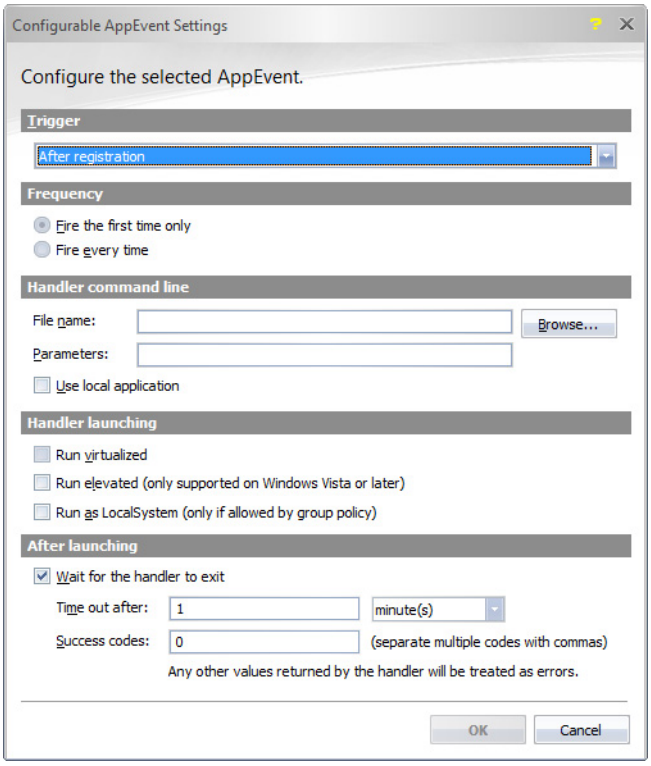


Figure 6-4: Configurable AppEvents Settings Dialog

3. Select a **Trigger** from the drop-down menu. A *trigger* is condition that causes an *handler* to start, such as running a VB script.

AppEvent Trigger	Description
After registration	Occurs once when the application is first added
After activation	Occurs once immediately after an appset has been activated
After virtualization	Occurs immediately after virtualization has completed
After launch	Occurs immediately after the application launches
After exit	Occurs after the application exits
Before deactivation	Occurs once immediately before an appset deactivates
Before devirtualization	Occurs before devirtualization starts

4. Select the **Frequency** setting from the drop-down menu. *Frequency* is how many times an action is set to occur after the AppEvent is triggered.

Frequency	Description
Fire Once	The action will run the first time the AppEvent is triggered, and never again.
Fire Always	The action will be triggered each time the AppEvent occurs. Some triggers only occur once and this open will be disabled.

5. Select a **Handler command line** by clicking **Browse** to locate the name of the executable or script file, then select it.

   The file name is displayed in the **File Name** field.

NOTE

See About AppEvent Scripts and Executables on page 61 for important information about AppEvent scripts and executables.

6. Enter any necessary **Parameters**, which may include configuration switches or numeric values. For example: “visible = true” or “16” or “/faststart.”
7. Select **Use local application** if the Handler executable or script is located on the local PC or on the company network. If this option is selected, the handler is not included with the appset (**.stp** file).

CAUTION

Use local application can be used to specify an executable within the appset. However, there are some limitations. The triggers can only be “After virtualization”, “After launch”, “After exit”, or “Before devirtualization”. The handler will always be launched virtualized (regardless of “Run virtualized” setting).

8. Choose how the handler is to be launched:
   - **Run virtualized** - This option specifies if the handler should see isolated assets (i.e. files and registry keys) or not. It can only be specified for triggers that occur after an application has been virtualized.
   - **Run elevated** - This option specifies if the handler should be launched elevated requiring administrator privileges. It only affects handlers that execute on Windows Vista or later.
   - **Run as LocalSystem** – This options specifies that the handler should be launched as a SYSTEM process. This means the handler has access to files that the current user also does. Handlers run as LocalSystem can affect critical files and should be used cautiously. This option will work only if the you have configured the Group Policy for Jukebox Studio to allow it.
9. Select the **After launching** options:
   - **Wait for handler to exit** - Requires Jukebox Player to wait for the handler to finish before continuing to the next state. If not set then the handler is executed in parallel with Jukebox Player continuing normal operations.
   - **Time out after** - Specify the amount of time, in seconds or minutes, Jukebox Player should wait for the handler to exit. If the handler has not returned by this amount of time, then the Jukebox Player will treat this as a failed return code.
   - **Success codes** - The handler must generate a specific *return value* to Jukebox Player that indicates either success or failure. By default 0 is the only success code and all other values are interpreted as a failure. However if 0 is not the desired then new success codes can be specified here.

   After launching options do not apply to the “post-launch” trigger as the application has already launched and cannot be interrupted.

CAUTION

Existing appset will use the default success codes of 0 for success and all other return values are interpreted as a failure.

10. Click **OK**. The AppEvent has now been added.

About AppEvent Scripts and Executables

An AppEvent handler is a script or executable program, which can be written in any scripting or programming language, such as Visual Basic, C++, etc.

NOTE

There is a 10 MB size limit for each script. Also, scripts can not be modified after they are cloudified into an STP file. Any modification will be seen as tampering and the script will be disabled.

Upon detecting a failed success code returned from the AppEvent handler, Jukebox Player will log a failure with the return code but will not display any messages for error codes. Therefore, when creating a script that will return an error code upon failure, it is necessary that the script itself displays any desired message to the user. If the failure is during the activation sequence, then the application will not be added to Jukebox Player. If the failure is during the removal process, then the removal will simply continue.

NOTE

During a Windows logoff or shutdown, AppEvent triggers will be skipped because Windows does not allow new process to be created during this time.

Several sample Visual Basic AppEvent scripts are included with Jukebox Studio. These sample scripts check the local system for the existence of the applications shown in the table below.

Script File Name	Description of AppEvent Script	Error Code
ReaderCheck.vbs	Checks for existence of **Adobe Reader**	**1**
QuicktimeCheck.vbs	Checks for existence of **QuickTime**	**1**
DotNetCheck.vbs	Checks for existence of **.NET Framework**	**1**
DirectXCheck.vbs	Checks for existence of **DirectX**	**1**
AddService.exe	Used to add and start a service	**-1**
RemoveService.exe	Used to stop and remove a service	**-1**
OSMemCheck.vbs	Verifies there is enough memory	**1**
OpenFirewall.vbs	Open firewall for programs or ports	**-1**

NOTE

When using the sample Visual Basic scripts included with Jukebox Studio, make sure to enter the **Parameters** setting in the Configurable AppEvent Settings dialog with the **major** and **minor** numbers of the required software version. For example, to detect **QuickTime version 7.3** use a **Parameters** setting of **7.3**. Optionally a title can also be set for the failure dialog (e.g. QuicktimeCheck.vbs 7.3 QuickTime)

About AppEvent Context

An AppEvent has access to environment variables that define specific information for the application being paged. An AppEvent can query this information to be used as needed.

- AppName - The full name of the application
- AppID - The Application ID for this appset.
- WorkingDir - The **Working folder** set under the advanced section below.
- CommandLine - The **Command line** set under the advanced section below.
- ClientId - The unique Player ID used by the Jukebox Server to associate with a user names.

- LastAccessTime - The last time this application was launched if set.
- ProviderUrl = URL to the Jukebox Server or SaaS provider portal if set.
- SessionKey - Option token parameter set by a SaaS portal.
- Extra - User defined field set by a SaaS portal. Please see the *Application Jukebox Server Administration Guide* for more details.

CAUTION

When using the Extra parameter, be sure to not include the <, >, |, &, ^, “, #, %, {, }, \, ~, [,], ‘, or space characters unless they are encoded by the portal and decoded by the handler. All characters will appear as part of the value in the extra environment variable and do not need to be escaped.

Virtualization Settings

The Virtualization setting controls how the cloudified application interacts with the Target PC running Jukebox Player and whether specific parts of the application are visible to the whole system or just the cloudified application.

To access **Virtualization Settings** dialog, click the **Settings** menu and select **Virtualization**. The Settings dialog appears with the **Virtualization** screen visible.

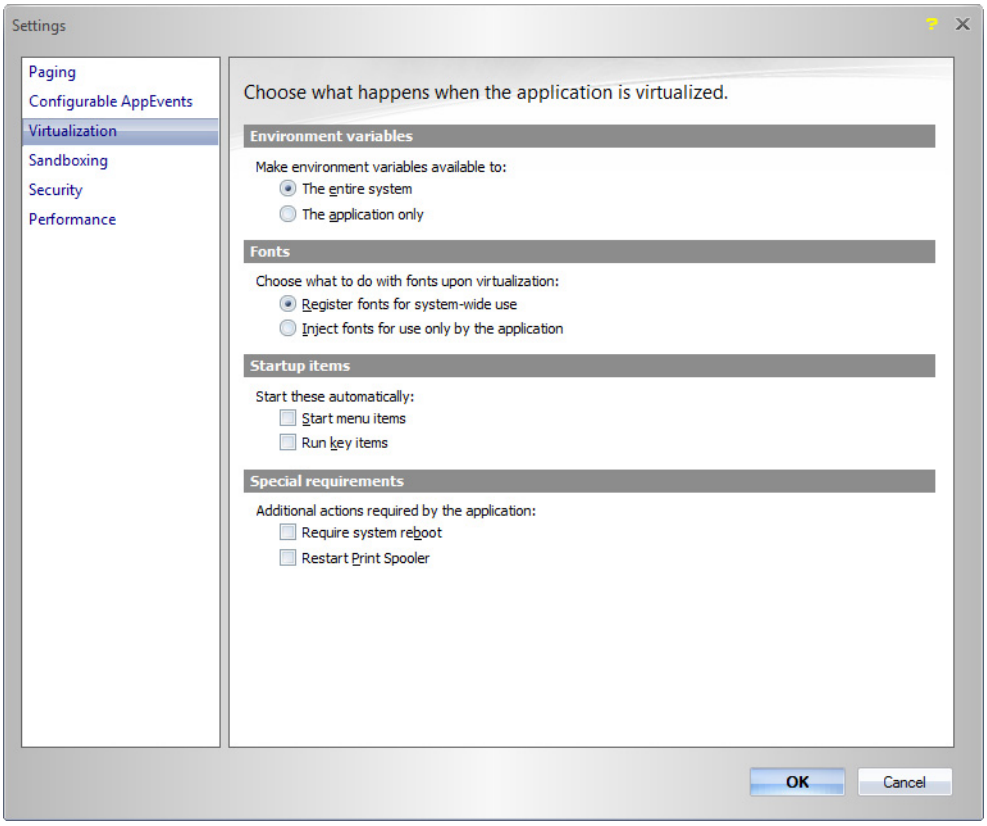


Figure 6-5: Settings Dialog, Virtualization settings

- **Environment Variables** – There are two options. You can make environment variables in the appset available to:

The entire system – Select this option if you want the environment variables provided with the appset to be shared with all applications installed on the Target PC.

The application only – Select this option if you want the environment variables provided with the appset to only be used by the cloudified application. Other applications installed on the Target PC will not be able to access these environment variables.

- **Fonts** – There are two options:

  Register fonts for system wide-use – Select this option if you want the fonts provided with the appset to be shared with all applications installed on the Target PC.

  Inject fonts for use only for this application – Select this option if you want the fonts provided with the appset to only be used by the cloudified application. Other applications installed on the Target PC will not be able to access these fonts.

NOTE

When selecting **Inject fonts...**, it is recommended to also set the disposition of the **SystemFonts** folder to virtual-isolated (layer-4).

- **Startup Items** – There are two options:

  Start menu items – Select this option if there are Startup items for the application under the Start menu that should be launched after virtualization.

  Run key items – Select this option if there are Run registry key items for the application under HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run that should be launched after virtualization.

- **Special requirements** – There are two options:

  Require system reboot: Select this option if you want the user to reboot their PC after virtualization. This is only needed if the application requires a reboot in order to run properly, such as to setup services.

  Restart print spooler: Select this option if there is a printer driver that needs to be setup after virtualization by restarting the printer spooler service.

Sandboxing Settings

The act of *sandboxing* means to separate changes made to the file system or registry from the real system on the Target-PC. Application Jukebox Studio performs partial-sandboxing for files, which means that only content modified or added to folders or sub-folders of root folders in the project are sandboxed. For example, if a project contains only the installation root folder and “Program Files,” and if the cloudified application creates a new file under the “Windows” folder, then this new file will be directly written to the system. Application Jukebox Studio will sandbox all default registry hives (i.e. HKCR, HKCU, HKLM, HKU), but any other hive must be part of the appset to be sandboxed.

For the paths in the project that are sandboxed, the content is separated from the system, meaning that they are not written to the local system but stored separately and only made available to the paged application. Again using the example above, if a new file under “Program Files” is created, then it will be sandboxed and not written to the system. Only existing files under “Program Files” that are not contained in the appset would be modified by the application if attempted. The same rule applies for existing registry keys not contained within the appset.

Sandboxing is not intended to be applied to documents or data saved by the user using the paged application. To help avoid this, data written to the following folders are the only exception for sandboxed folders:

Sandbox Exclusion Folder	Default Path - Vista
System Root	C:\
Desktop	C:\Users\<username>\Desktop
Documents	C:\Users\<username>\Documents
Downloads	C:\Users\<username>\Downloads
Music	C:\Users\<username>\Music
Pictures	C:\Users\<username>\Pictures
Videos	C:\Users\<username>\Videos
Links	C:\Users\<username>\Links
Favorites	C:\Users\<username>\Favorites
Contracts	C:\Users\<username>\Contacts
Saved Games	C:\Users\<username>\Saved Games
Searches	C:\Users\<username>\Searches
Temp	C:\Users\<username>\AppData\Local\temp
Internet Cache	C:\Users\<username>\AppData\Local\Microsoft\Windows\Temporary Internet Files
Shared Documents	C:\Users\Public\Documents
Shared Downloads	C:\Users\Public\Downloads
Shared Music	C:\Users\Public\Music
Shared Pictures	C:\Users\Public\Pictures
Shared Video	C:\Users\Public\Videos

Data written to those paths is directly written to the system, even if the “Users” folder is part of the appset as a root folder, as would be expected by end-users.

All the sandboxed content is deleted when the application is removed. To preserve the application settings (e.g. changes to a toolbar and other UI customizations) even after removing an application, select the **Preserve application changes** option in the **Sandboxing** screen of the **Settings** dialog.

To access the **Sandboxing** settings, click the **Settings** menu and select **Sandboxing**. The Settings dialog appears with the **Sandboxing** screen visible.

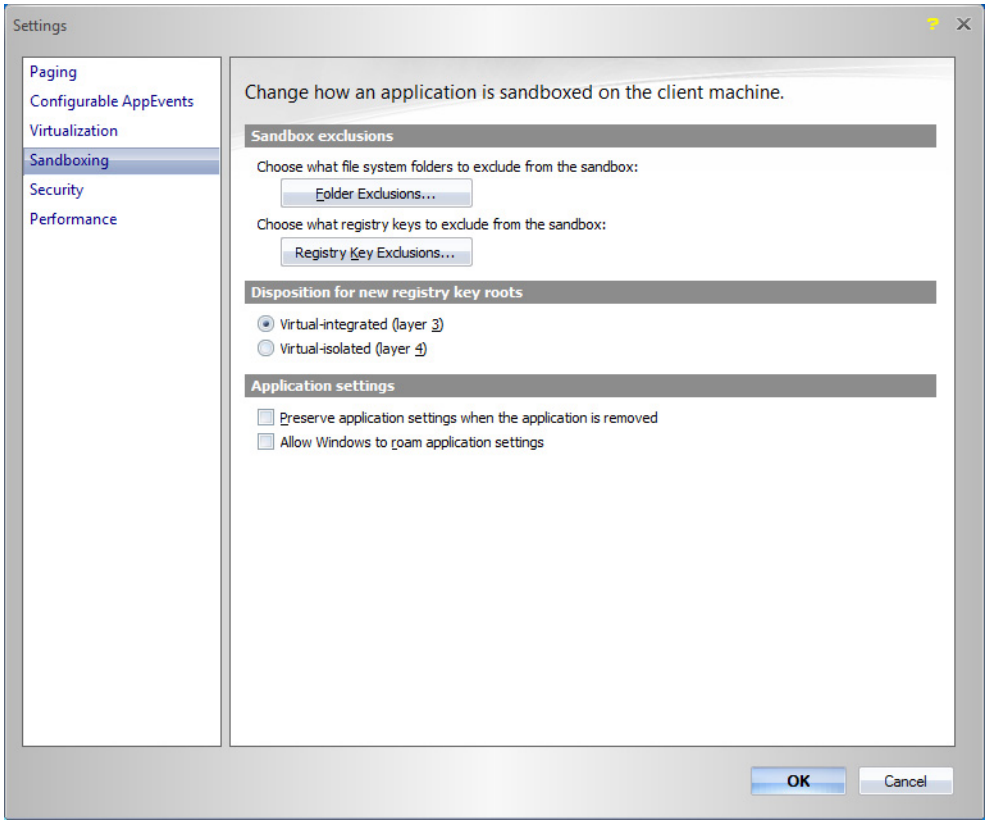


Figure 6-6: Settings Dialog, Sandboxing Screen

- **Sandbox Exclusions –** If necessary, specific registry keys and folders can be excluded from the sandbox. This may be required if the asset in question is shared between a cloudified application and another locally installed application.

To add a folder exclusion

1. Click **Folder Exclusions**.
2. The **Sandbox Folder Exclusions** dialog appears.
3. If the excluded folder exists on the Cloudifying PC, you can drag the folder from Windows Explorer into the **Sandbox Folder Exclusions** dialog.
4. If the excluded folder does not exist on the Cloudifying PC, click **Add...**
5. A blank line will appear on the **Sandbox Folder Exclusions** dialog. Type the full folder path and hit **Enter**.

To add a registry key exclusion

1. Click **Registry Key Exclusions**.
2. The **Sandbox Registry Key Exclusions** dialog appears.
3. Click **Add...**
4. A blank line will appear on the **Sandbox Registry Key Exclusions** dialog. Type the path and hit **Enter**.

- **Default Disposition for new registry key roots –** Registry key roots created dynamically by the virtualized application while running will use the selected disposition:
  - **Virtual-integrated (layer 3)**
  - **Virtual-isolated (layer 4)**

NOTE

This setting replaces part of the **Virtualization Settings** option in previous versions of Jukebox Studio. The rest of this functionality is found under the **Global Disposition Settings** dialog. See Globally Changing Asset Dispositions on page 113 for more information.

- **Application Settings –** Any application-specific settings, such as recently opened files, will be lost when the application is removed from the Jukebox Player unless one of the following options is selected.

  Preserve application changes: Select this option if you want the application-specific settings modified by the user to be preserved after the application is removed from Jukebox Player. If the application is added back to Jukebox Player, then these application settings will be used.

  Roam application changes: Select this option if you want the application-specific settings to be written out to the standard “Windows Roaming Profiles” location. Windows can then roam the settings if configured to do so on the Jukebox Player PC.

Security Settings

There are times when the standard file security settings (see Editing Folder Properties on page 77) need to be bypassed by specific application processes. For example, an application patch (such as Windows Updates) may be required to modify a file that is protected with the “Make read-only” flag. In this case, a *security override* can be used to allow this to happen while still protecting the application. Another example would be to not allow anti-virus or indexing software from fetching all the application pages. This undesirable side-effect can also be prevented by denying those specific processes from accessing the virtual application.

NOTE

Security overrides only affect the file security settings but do not change the visibility of the disposition layer. If a file is layer-4, then setting a system process to override all the security settings will still not allow that process to see the isolated layer-4 file.

To set up a security override:

1. To access the **Security** settings, click the **Settings** menu and select **Security**. The Settings dialog appears with the **Security** screen visible.

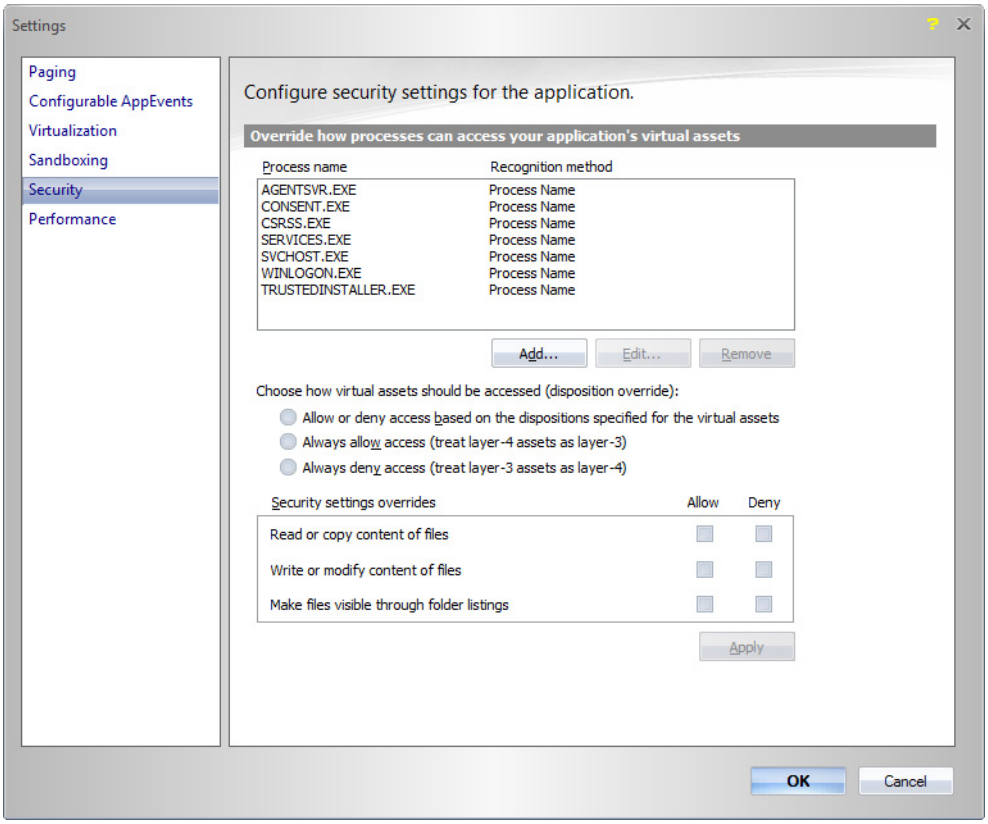


Figure 6-7: Settings dialog, Security Overrides screen

NOTE

For advanced users: To learn how to modify the initial settings for this dialog, see **Appendix B**, Editing Configuration Files on page 167.

2. Click **Add** to add a process name to the security override list. This will open the Security Override Process dialog.

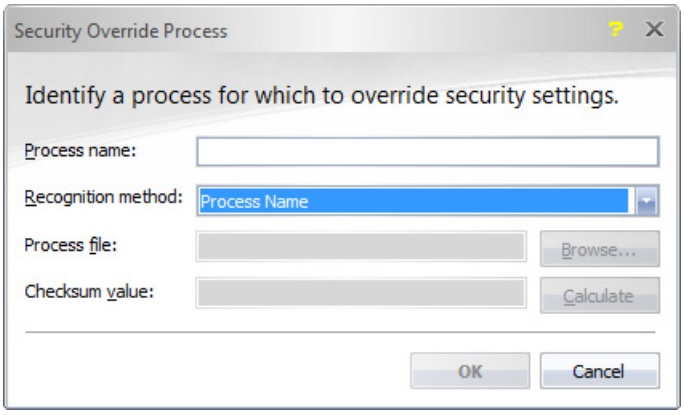


Figure 6-8: Security Override Process Dialog

3. Set the following information and click **OK** when complete:

- **Process name** – The exact name of the process for the security override controls. This must be the full name and wildcards may not be used. This field will be auto-populated when using the Process File browse button.
- **Recognition method** – There can be duplicate process names in the list that correspond to distinct file versions (e.g. multiple patch executables all with the same name). In order to distinguish between them, a recognition method must be used to authentic the process. The following types are supported:
  - Process name - Weakest method that only verifies the process name as specified.
  - Process path - Verifies the process name and path, as specified, is where the process file is located on the local system.
  - CRC Checksum - Uses a CRC algorithm to compute a checksum of the executable and stores it to quickly check that the process files is authentic.
  - MD5 Hash - Uses a MD5 algorithm to compute a fingerprint of the executable and stores it to verify that the process file is authentic. This is the recommended method to use as it is the most secure.
- **Process file** – The full path to the process file.
- **Checksum value** – Used by the recognition method to store a fingerprint of the process file based on the method selected.

4. Select whether to override the disposition settings for virtual assets.
   - Allow or deny access based on the dispositions specified for the virtual assets – This setting causes the selected process to respect the individual disposition settings for files, folders, and registry keys/values.
   - Always allow access – This setting causes the selected process to treat all virtual asset as if they were virtual-integrated (layer 3), even if they are set at virtual-isolated (layer 4).
   - Always deny access – This setting causes the selected process to treat all virtual asset as if they were virtual-insolated (layer 4), even if they are set at virtual-integrated (layer 3)
5. Next set the process Allow and Deny access control and choose from the following:
   - **Write** – Sets permission to write, delete, or create all of the application's files.
   - **Read** – Sets permission to read all of the application's files.
   - **Find** – Sets permission to search for any of the application's files.
   - **Execute** – Sets permission to execute any of the application's files.

NOTE

The **deny** settings will always take precedence over the **allow** settings, which means that setting both will result only in the deny action.

6. Click **Apply** to set the new security override settings for the specified process.

Multiple processes can be added to the security override list by repeating the steps. Each process can also be modified or removed as needed.

Performance Settings

The **Performance** screen control what percentage of the cloudified application can be stored temporarily in the local Jukebox Player cache. For more information see Adjust Cache Size on page 52.

To access the **Performance** settings, click the **Settings** menu and select **Performance**. The Settings dialog appears with the **Performance** screen visible.

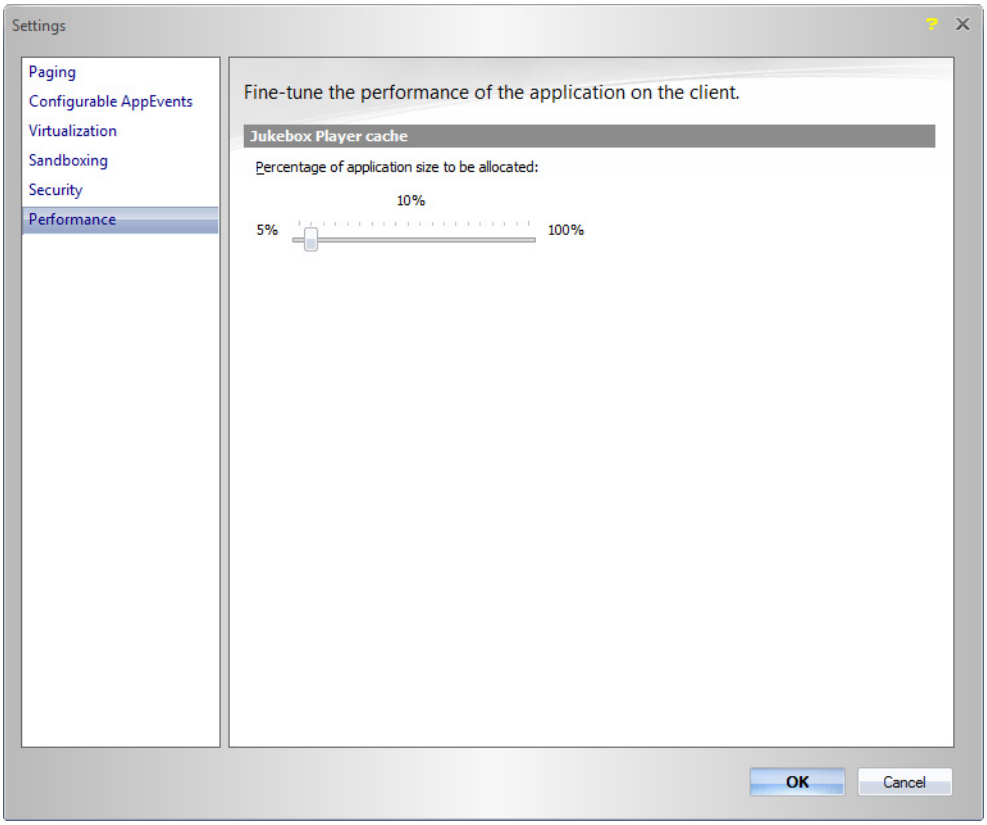


Figure 6-9: Settings Dialog, Performance screen

- **Jukebox Player cache** – The percentage entered here is the required amount of cache space the application will need to run smoothly. In most cases, the default of 10% is sufficient. This space will be allocated to the minimum size within the Jukebox Player cache. Setting this field may cause the Jukebox Player to prompt to increase a user's cache minimum size. After testing and optimization is completed in Phase 4, this additional space can be adjusted accordingly if performance was deemed poor with the default amount.

Chapter 7

Modifying Project Assets

The following are explained in this chapter:

- **Modifying Files and Folders**
- **Modifying Registry Entries**
- **Modifying Environment Variables**
- **Modifying Fonts**
- **Modifying Startup Items**
- **Modifying Services and Drivers**
- **Searching for Assets**

Overview

At various times it may be necessary to make modifications to the project, including adding, deleting, or changing appset assets, or removing *artifacts* (unwanted DLLs, etc.) that were introduced as a side effect of capturing the application installation. This can be easily done using the various view tabs in Jukebox Studio: **Files**, **Registry**, **Environment**, **Fonts**, **Startup**, and **Services & Drivers**.

NOTE

The material in this chapter applies to “cleaning” up the appset (Step 5) in **Chapter 3**, Capturing the Application Installation on page 27, preparing an appset for cloudifying in **Chapter 4**, Cloudifying the Application on page 43 and to modifying a patched appset in **Chapter 10**, Patching an Appset on page 157.

Modifying Files and Folders

The Files view is accessed by selecting **Files** from the **View** menu, or by selecting the **Files** view as shown below.

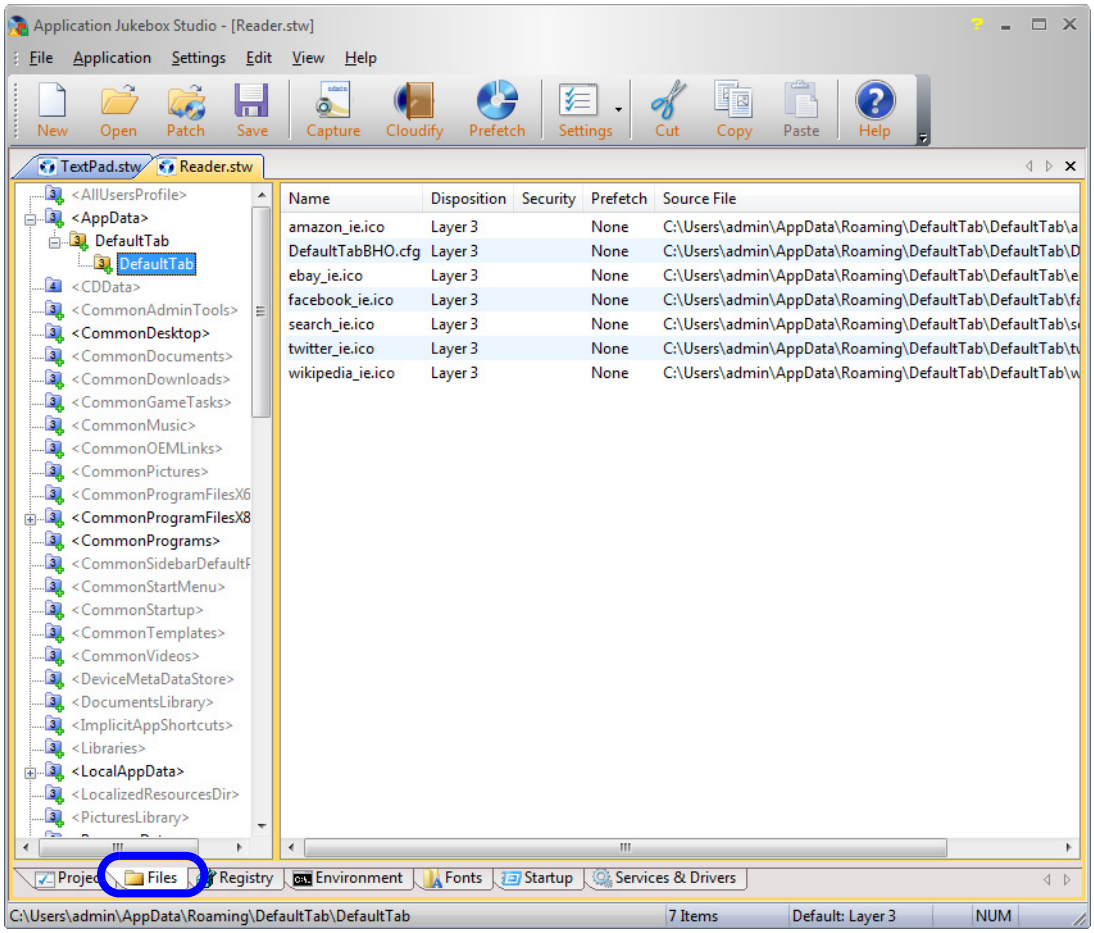


Figure 7-1: Files view

Most commands can be accessed either from the **Edit** menu or by right-clicking a folder in the folders (left) pane, or by right-clicking a file in the files (right) pane.

NOTE

Security descriptors are single letters based on the keyboard shortcuts in the File property dialog. For example, R = *Make **R**ead-Only*.

About Folders

There are the two types of folders that appear in the folders (left) pane of the Files view. These folders are colored to make identification simple. All the parent folders in the view are considered “root” folders to the project. These root folders correspond to various locations on the real file system.

Root and Template Folders

- **Root folders** – The yellow-colored folders are folders that were captured during installation or manually added to the top of the folder pane (left) of the Files tree view. To avoid folder conflicts on the Target PC, it is recommended to install the applications being captured to a Root folder, and to give this folder a unique name.
- **Template folders** – All the blue-colored folders have unique names called *templates* that correspond to *special folders* in Windows. *Templates* are aliases that map to actual system folder paths (e.g. *WindowsDir* typically maps to “c:\windows”). The specific system folder that a template maps to depends on the version of the Windows operating system, the particular operating system installation (e.g. if Windows is installed on the D drive, then *WindowsDir* will map to “d:\windows”), the logged-in user, or a combination thereof. So templates allow for the Target PC to have root folders in different locations, making it possible for an appset to “adapt” to the Target PC (e.g. if a user moves the “My Documents” folder to a different location, the corresponding template will map to that new location when the appset is initially paged). Template folders cannot be removed or added to the view. Empty templates folder (shown with gray text) will not be added to the final cloudified application.

NOTE

For a listing of supported Template Mappings for folder, please see the Appendix. More information on folder mappings (called “special folders” under Windows) is available online at http://support.microsoft.com/kb/194702/en-us.

CDData is not a special folder in Windows, but is treated like one by Application Jukebox Studio. If content is placed within this folder, then a virtual CD device will be created on the Target PC with the folder’s content. This is used to simulate CD/DVD data being present on the system.

CAUTION

The “Documents and Settings” folder is not a system root folder since Windows does not support a template mapping for it. Because of this, be sure this folder is **NOT** included in the appset as an Installed Folder. The sub-folders in “Documents and Settings” all have templates.

Merged and Non-merged Folders

Merged folders are identified by a green plus sign (“+”) over the folder icon. These are typically folders that exist on a real system. Files or sub-folders within these folders will be *merged* with the contents on the Target PC during the virtualization phase. If there is a name conflict with a real sub-folder or file on the Target PC, only the paged version will be visible. These folders will also be merged with corresponding folders in other paged applications.

Non-merged folders set to virtual-integrated (layer-3) are still merged from the Target PC’s system view. However, from the virtual application’s point-of-view, the folder and content are not merged. The virtual application will only see the contents of these unmerged folders, ignoring the contents of the folder on the Target PC and merged folders from any other appsets. Non-merged folders set to virtual isolated (layer-4) are invisible to everything except the virtual application.

The examples below will help clarify these concepts.

Merged Folder Visibility

This section illustrates how an appset's merged folders get merged with the real folders of the Target PC and the folders of other appsets. The system will always have a merged view on content that it has access to (e.g. layer 3 content). Merged rules only apply to the virtual applications.

Suppose that appsets A, B, C and D have been paged on the Target PC, in that order, and that

- the system has a real folder X;
- appset A has a virtual-integrated (layer 3) merged folder X;
- appset B has a virtual-integrated (layer 3) merged folder X;
- appset C has a virtual-isolated (layer 4) merged folder X;
- appset D has a virtual-isolated (layer 4) non-merged folder X;
- appset E has no folder X.

For the purpose of this discussion, we only consider the case where the folder X contents in appsets are virtual-integrated (layer 3) or virtual-isolated (layer 4) since layer 1 and layer 2 content is treated just like system content. For more information about disposition layers see Editing Folder Properties on page 77.

The table below shows the visibility of folder X's content to the system and the various appsets. Each list in the left column shows the visibility priority in case of conflicting content. For instance, the first row indicates that

- the system will see non-conflicting content from B, A, and itself;
- conflicting content from B is visible to the system and hides all other;
- content from A that conflicts with the system, but not with B is visible to the system and hides the system's own content.

This...	Sees X Content from...
System	B, A, System
Appset A	A, B, System
Appset B	B, A, System
Appset C	C, B, A, System
Appset D	D
Appset E	B, A, System

- conflicting content between A and B results in A viewing A's own content and B viewing B's content, while the system views B's content.

For information about making a folder merged see Making Root Folders Merged on page 79.

Adding Root Folders

To add a root folder:

1. Right-click any folder or on a blank area of the folders pane (left).
2. Select **Add Root Folder** from the pop-up menu. The **Add Root Folder** dialog appears.

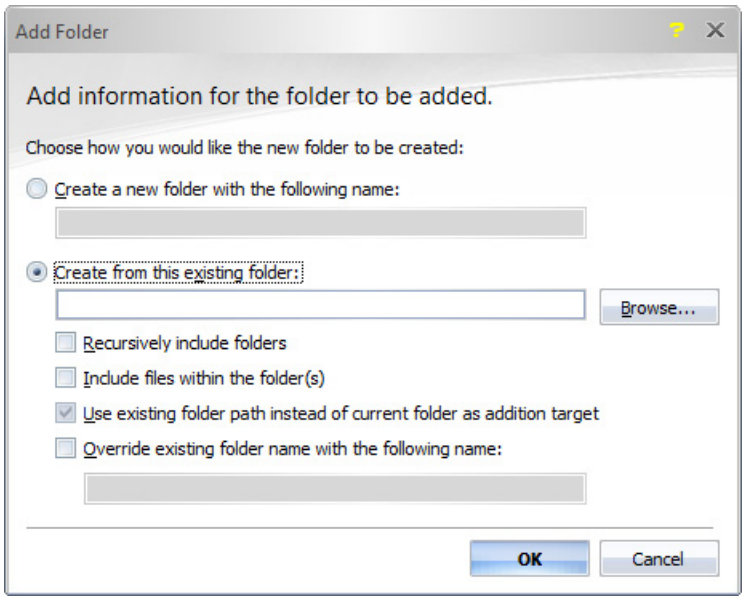

Add Folder
Add information for the folder to be added.
Choose how you would like the new folder to be created:
Create a new folder with the following name:
Create from this existing folder:
Browse...
Recursively include folders
Include files within the folder(s)
Use existing folder path instead of current folder as addition target
Override existing folder name with the following name:
OK
Cancel

Figure 7-2: Add Root Folder Dialog

3. If you wish to create a new root folder, select **Create new root folder with the following name** and enter a path for it. If you wish to add an existing folder instead, select **Create from existing source folder**.
4. If you chose **Create from existing source folder**, you can also select one or more of the following options:
   - **Recursively include subdirectories** – Selecting this option will cause all subfolders under the selected folder to be added as well. Otherwise, only the selected folder is added.
   - **Include files within the directory(s)** – Selecting this option will add all the files in the folders being added. Otherwise, only the folders are added.
   - **Use existing folder path instead of current folder as addition target** – This option is automatically selected because you are adding a root folder and not a subfolder.
   - **Override existing folder name with the following name** – Normally, the selected folder is added with its current name. If you wish to add the folder under a different name, clear this option and specify the new name to be given to the folder.
5. Click **OK**. The Files view will be updated to reflect the added folders and files.

Adding Sub-folders

To add a sub-folder

1. To add a sub-folder, right-click any root folder or folder template.

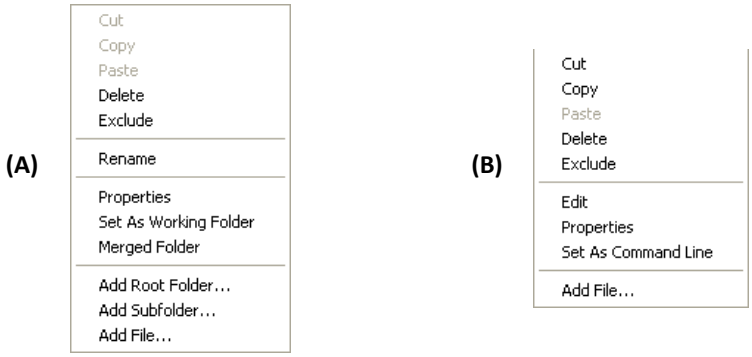


Figure 7-3: (A) Folder Options and (B) File Options Pop-Up menus

2. Select **Add SubFolder** from the pop-up menu. The **Add Folder** dialog displays.

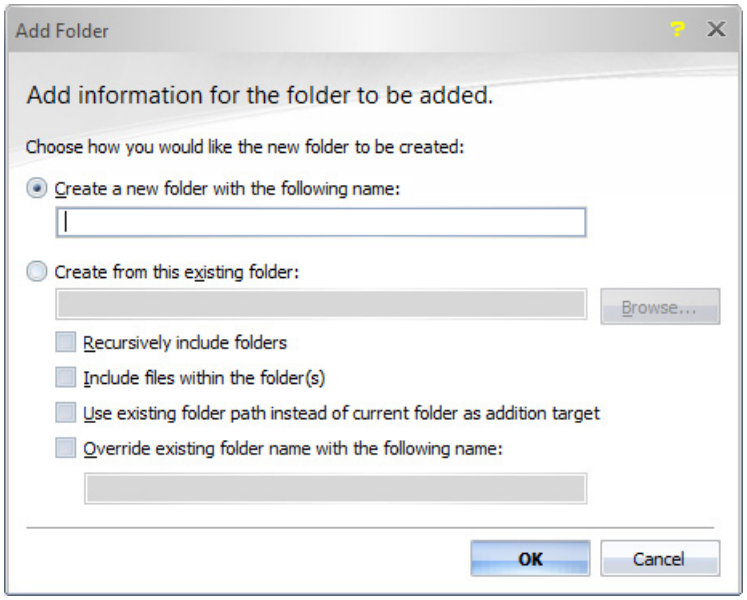

Add Folder
Add information for the folder to be added.
Choose how you would like the new folder to be created:
Create a new folder with the following name:
Create from this existing folder:
Browse...
Recursively include folders
Include files within the folder(s)
Use existing folder path instead of current folder as addition target
Override existing folder name with the following name:
OK
Cancel

Figure 7-4: Add Folder Dialog

3. If you wish to create a new folder, select **Create new folder with the following name** and enter a name for it. If you wish to add an existing folder instead, select **Create from this existing folder**.
4. If you chose to add an existing folder, you can also select one or more of the following options:
   - **Recursively include subdirectories** – Selecting this option will cause all subfolders under the selected folder to be added as well. Otherwise, only the selected folder is added.
   - **Include files within the directory(s)** – Selecting this option will add all the files in the folders being added. Otherwise, only the folders are added.
   - **Use existing folder path instead of current folder as addition target** – Instead of adding the new folder as a subfolder of the selected folder, the complete path of the new folder will be added to the project. This saves you from having to manual recreate the entire path, starting with the root folder.

- **Override existing folder name with the following name** – Normally, the selected folder is added with its current name. If you wish to add the folder under a different name, clear this option and specify the new name to be given to the folder.

5. Click **OK**. The Files view will be updated to reflect the added subfolders and files.

Renaming Folders

Any non-template folder can be renamed.

To rename a folder:

1. Right-click the folder that you wish to rename.
2. Select **Rename** from the pop-up menu.
3. Rename the folder and press **Enter** once you are finished.

Editing Folder Properties

A file has a number of security and prefetch settings that can be modified to suit your specific needs.

To edit folder properties:

CAUTION

For some of the settings discussed below, if **Recursively** is selected, all sub-folders and files within them will be affected.

1. Right-click the folder with properties you wish to change.
2. Select **Folder Properties** from the pop-up menu. The **Edit Folder Properties** dialog appears.

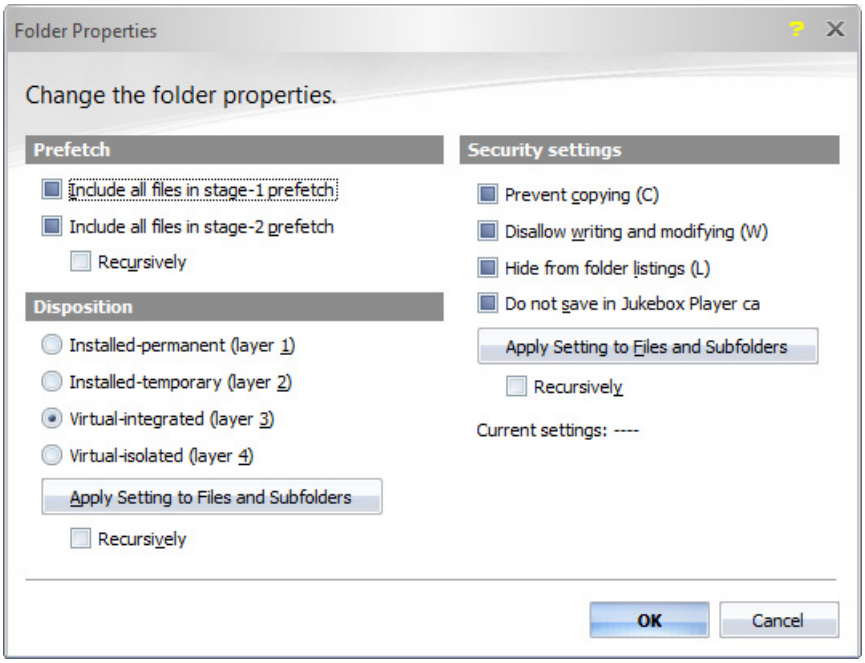


Figure 7-5: Edit Folder Properties Dialog

3. Configure the following settings as required:
   - **Prefetch** - Select **Include all files in Stage-1 prefetch** or **Include all files in Stage-2 prefetch**. For more information, see Prefetching on page 7.
   - **Disposition** - The layer number displayed on the folder indicates the disposition setting. Select one of these disposition settings:
     - **Installed-permanent (Layer 1)** - Copies assets (files and keys) permanently onto the local system, and can be seen by the entire local system.
     - **Installed-temporary (Layer 2)** - Installs assets during the activation process, and uninstalls assets during the deactivation process. The original asset is backed up before the new asset is installed, and when the new is uninstalled, the original asset is restored.
     - **Virtual-integrated (Layer 3)** - (default setting) Assets that can be seen by both the paged application and the local system, but are not physically installed on the local system.
     - **Virtual-isolated (Layer 4)** - Assets that can only be seen by the paged application and are not physically installed on the local system.

CAUTION

Layer 1 or layer 2 files cannot have security settings applied to them. In addition, Jukebox Player will not track usage against files that are layer 1 or layer 2.

NOTE

Layer 1 and layer 2 options are disabled for non-merged folders because folders with layer 1 or layer 2 dispositions are always merged. See Merged and Non-merged Folders on page 73

   - **Security Settings** - Select one of these settlings:

Security Setting	Description	File enabled by default
Deny Reading and Copying	Assets can not be copied	**Command-line file**
Deny Writing and Modifying	Assets can not be altered	**Command-line file**
Hide from folder listings	Assets are not displayed in listing of folders	**(none)**
Do not cache on local machine	Assets are not permitted be cached to the local system	**(none)**

4. Click **OK**. The Files view will be updated to reflect the changes made.

Setting the Working Folder

The **Working Folder** is the folder that contains the application's executable or some related files. A vast majority of the time, using the same directory as used in the command line will work. On occasion using the same directory as where the program executable is found will not work. In these cases, it may be required to set the Working Directory to the application root directory. The easiest way to populate this field is to navigate in the Files view to the folder that you want to set as the working folder, right-click it, then select **Set As Working Folder**.

Making Root Folders Merged

By default, root folders are virtual and their contents will not be merged with folders already on the system.
See Merged and Non-merged Folders on page 73

To merge an Root folder with system folder:

1. Right-click the Root folder you wish to change.
2. Check the **Merged Folder** option.
3. A green plus sign (“+”) will appear over the folder symbol.

Adding Files

Jukebox Studio allows you to add any files to any folder.

To add a file to a folder:

1. Right-click the desired folder or click a blank area of the Files view if the desired folder is currently selected.
2. Select **Add File** from the pop-up menu. The **Add File** dialog appears.

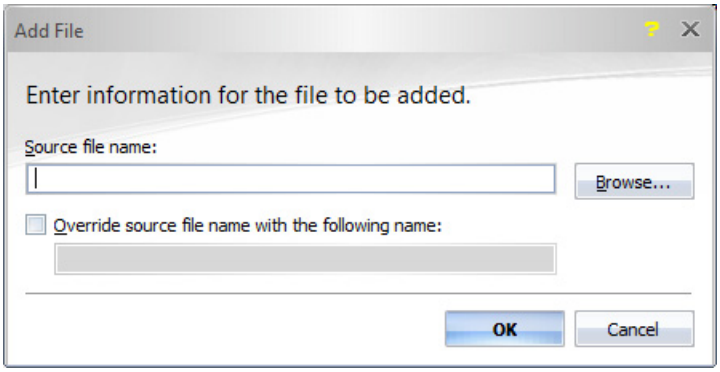


Figure 7-6: Add File Dialog

3. Enter the path of the file you wish to add or browse to it.
4. Normally, the selected file is added with its current name. If you wish to add the file under a different name, select **override source file name with the following name**.
5. Enter the new file name.
6. Click **OK**. The Files view will be updated to reflect the added file.

Editing Files

If after adding a file you realize that you have added the wrong file or you decide that you wish to change its name, you can do so without the need to delete the file and add it again.

To change the source or name of a file:

1. Right-click the file whose source or name you wish to change.
2. Select **Edit File** from the pop-up menu. The **Edit File** dialog appears.

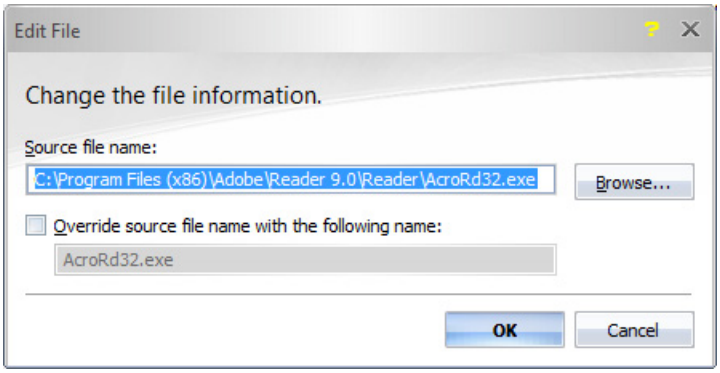


Figure 7-7: Edit File Dialog

3. To change the source of the file, enter the new path of the source file or browse to it.
4. To change the name of the file, make sure the **Override source file name with the following name** option is not selected, and specify the new file name.
5. Click **OK**. The Files view will be updated to reflect the changes that were made.

Editing File Properties

A file has a number of security and prefetch settings that can be modified to suit your specific needs.

To edit file properties:

CAUTION

For some of the settings discussed below, if **Recursively** is selected, all sub-folders and files within them will be affected.

1. Right-click the file that has properties you wish to change.
2. Select **File Properties** from the pop-up menu. The **Edit File Properties** dialog appears.

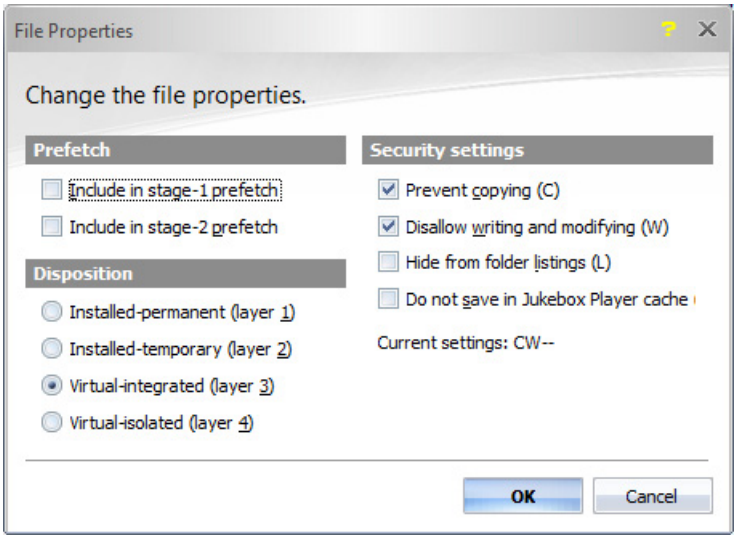


Figure 7-8: Edit File Properties Dialog

3. Configure the following settings as required:
   - **Prefetch** - Select **Include in stage-1 prefetch** or **Include in stage-2 prefetch**. For more information see Prefetching on page 7.
   - **Disposition** - See Editing Folder Properties on page 77.
   - **Security Settings** - See Editing Folder Properties on page 77.

CAUTION

Security settings are **not** enabled by default. Please add them to files you wish to protect.

Deleting Files and Folders

It is possible to delete any file or root folder that was created during installation or that was added manually.

To delete a file or folder:

1. Right-click a file or folder that you wish to delete.
2. Select **Delete** from the pop-up menu.
3. Select **OK** when prompted for confirmation.
4. If you wish to delete another file or folder, repeat steps 1-3.

Excluding Files and Folders

Excluded folders, sub-folders, and files are not included when the appset is cloudified, but are still part of the project. This gives you the flexibility of including certain folders and files for one deployment of an appset and not for another. Excluding folders and files also enables you to test an appset without a folder or file before permanently deleting it from the project.

To exclude a file or folder:

1. Right-click a file or folder that you wish to exclude.
2. Select **exclude** from the pop-up menu.
3. Excluded folders will have a red X over their symbol. Excluded files are shown with their name and properties in italicized text.
4. If you wish to exclude another file or folder, repeat steps 1-3.

Setting the Command Line

The **Command Line** is the path and executable file that are used to start the program.

To set the Command Line:

1. In the Files view, navigate to the application's executable, and right-click it.
2. Select **Set as command line**.

- If you are not sure where the executable is located, one way it can be found is by looking at the properties of the shortcut created when the application was installed.
- If you want to launch a **non-.exe** file such as a **.doc** file, it is possible. To cloudify non-applications, the command line must be set to specify an executable to be launched (such as an .exe or .com file). Then the name of the data file can be specified. For example: *"%SystemRoot%\System32\mspaint.exe" "C:\My Images\Image.bmp"*.

Locating Files/Folders in Windows Explorer

If for any reason you need to examine a file or folder in its physical location on the Cloudifying PC, Jukebox Studio can display the file or folder in Windows Explorer directly from the Files view.

To show a file in Explorer:

1. Right-click on the file asset in the right-hand pane of the Files view.
2. Select **Show Source File in Explorer**.
3. Windows Explorer will open to the correct folder with the selected file highlighted.

To show a folder in Explorer:

1. Right-click on the folder in the left-hand pane of the Files view.
2. Select **Show in Explorer**.
3. Windows Explorer will open to the correct folder.

Modifying Registry Entries

The Registry view is accessed by selecting **Registry** from the **View** menu, or by selecting the **Registry** view as shown below.

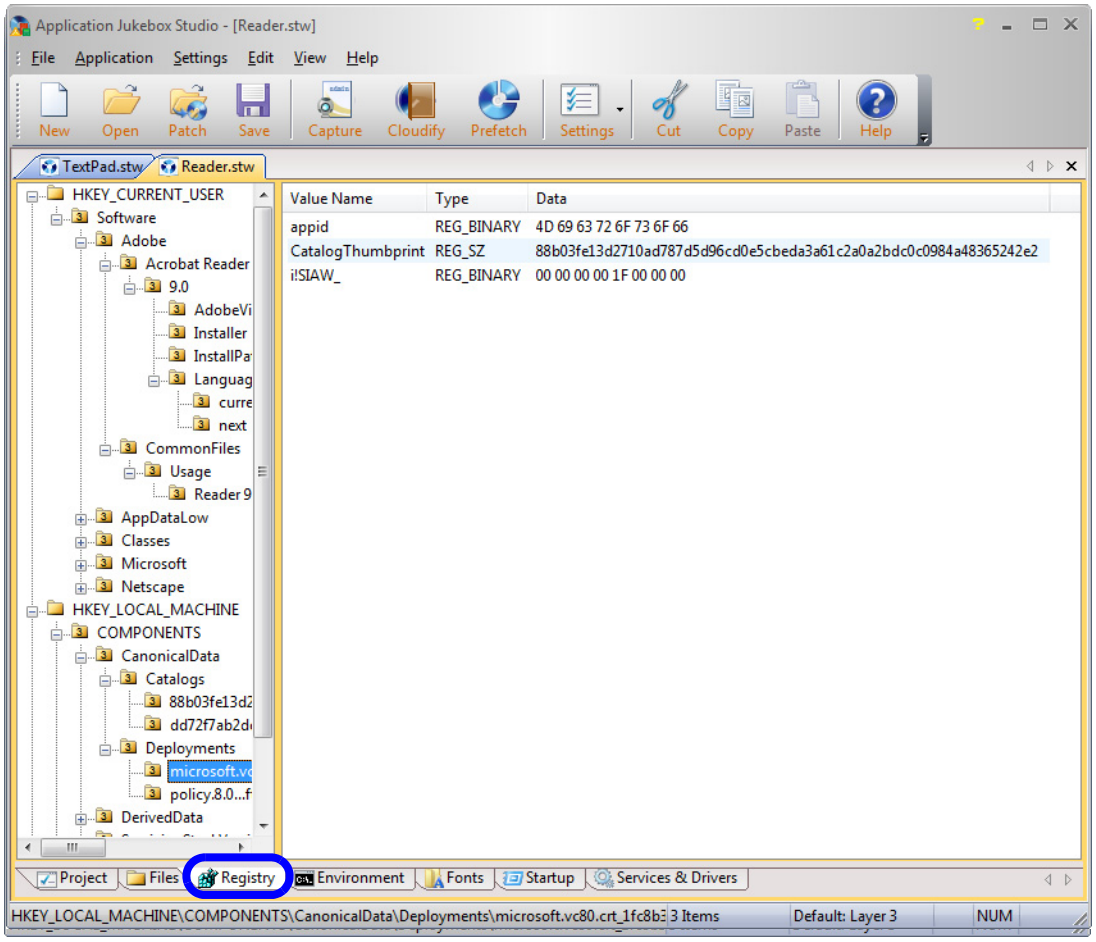


Figure 7-9: Registry view

Most commands can be accessed either from the **Edit** menu or by right-clicking a key in the registry keys pane (left), or right-clicking a value in the registry values pane (right).

Adding Registry Keys

You can create a new subkey or add an existing subkey under any key in the Registry view, including *root keys* (or *hives*). However, you cannot create new root keys.

To add a subkey

1. Right-click any key in the registry keys pane (left).
2. Select **Add Key** from the pop-up menu. The **Add Registry Key** dialog appears.

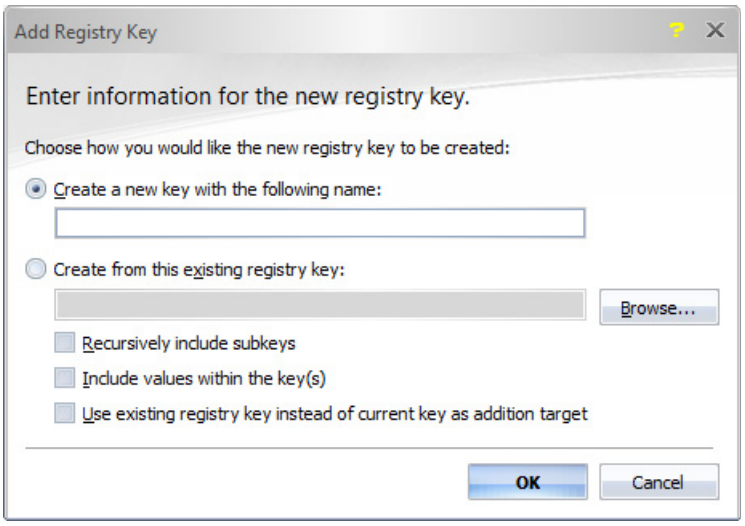

Add Registry Key
Enter information for the new registry key.
Choose how you would like the new registry key to be created:
Create a new key with the following name:
Create from this existing registry key:
Browse...
Recursively include subkeys
Include values within the key(s)
Use existing registry key instead of current key as addition target
OK
Cancel

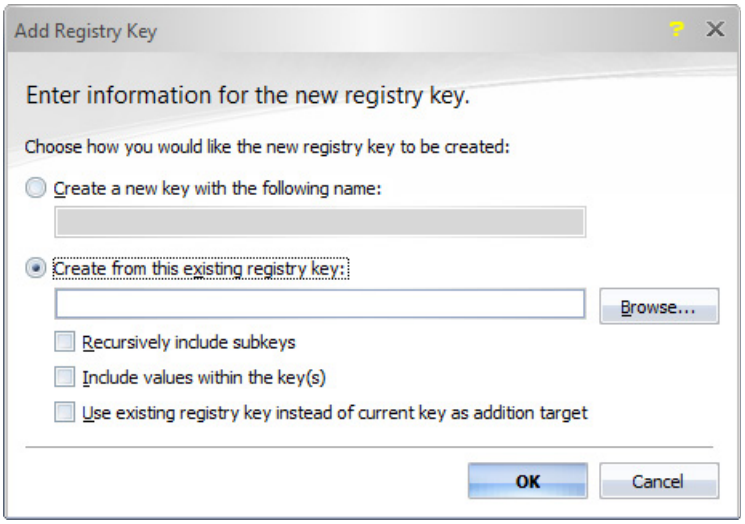
Figure 7-10: Add Registry Key Dialog

3. Enter a name for the new subkey. If you wish to add an existing key instead, select **Create from this existing registry key**.
4. Click **OK**.

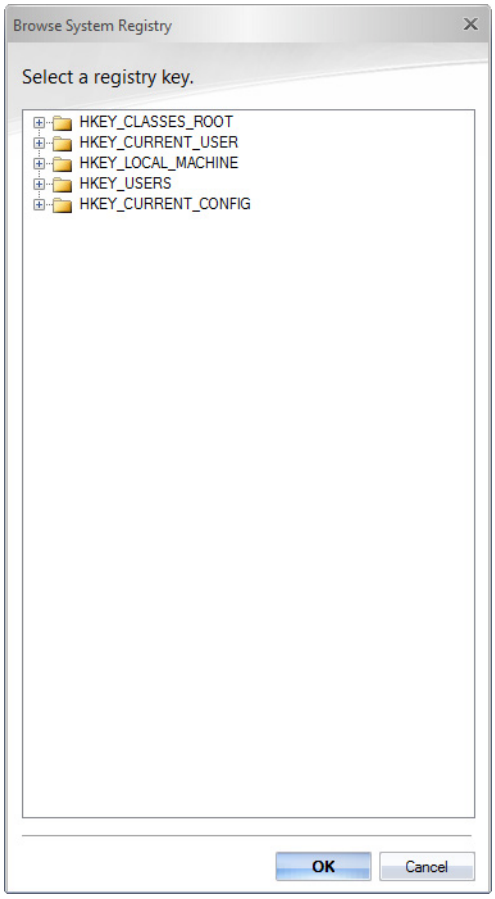
   The Registry view will be updated to reflect the added subkeys and values.

To add an existing subkey

1. Right-click any key in the registry keys pane (left).
2. Select **Add Key** from the pop-up menu. The **Add Registry Key** dialog appears.


Add Registry Key
Enter information for the new registry key.
Choose how you would like the new registry key to be created:
Create a new key with the following name:
Create from this existing registry key:
Browse...
Recursively include subkeys
Include values within the key(s)
Use existing registry key instead of current key as addition target
OK
Cancel

3. Select Create from this existing registry key.
4. Click **Browse...**
5. The Browse System Registry dialog appears.



6. Locate and select the desired key.

7. Click **OK**.
8. You can then select one or more of the following options:
   - **Recursively include subkeys** – Selecting this option will cause all subkeys under the selected key to be added as well. Otherwise, only the selected key is added.
   - **Include values within the key(s)** – Selecting this option will add all the values in the keys being added. Otherwise, only the keys are added.
   - **Use existing registry key instead of current key as addition target** – Instead of adding the new key as a subkey of the selected key, the complete path of the new key will be added to the project, starting from top level. This saves you from having to manually recreate the entire path, starting with the top level.
9. Click **OK**.

Renaming Registry Keys

To rename a registry key

1. Right-click the registry key that you wish to rename.
2. Select **Rename** from the pop-up menu.
3. Rename the folder and press **Enter** once you are finished.

Changing the Disposition of a Registry Key

To change the disposition of a registry key:

1. Right-click the desired key, and select **Disposition** from the pop-up menu.

   The **Registry Disposition** dialog appears.

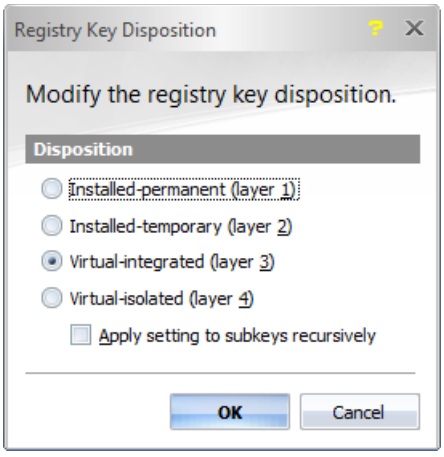


Figure 7-11: Registry Disposition Dialog

2. Select the disposition. For more information about disposition settings, see Editing Folder Properties on page 77.
3. If you want that disposition to apply to all subkeys, click **Apply setting to subkeys recursively**.
4. Click **OK**.

The Registry view will be updated to reflect the changed disposition setting for the selected key.

Adding Registry Values

Jukebox Studio allows you to add values to any key.

To add a registry value to a registry key:

1. Right-click the desired key or on a blank area of the Registry values pane if the desired key is currently selected.
2. Select **Add Value** from the pop-up menu. The **Add Registry Value** dialog appears.

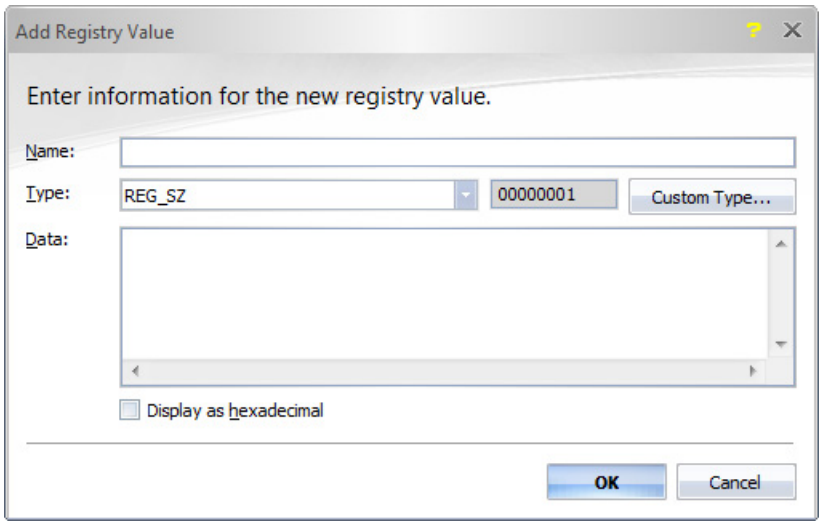


Figure 7-12: Add Registry Value Dialog

3. Enter the name of the value you wish to add in the **Name** field.
4. Select the data type of value from the **Type** drop-down list.
5. If you wish to view or enter the data in hexadecimal format, check the **Display as hexadecimal** option. Otherwise, data is assumed to be in decimal format.
6. Enter the data in the **Data** field.
7. Click **OK**. The Registry view will be updated to reflect the added value.

NOTE

If the registry value appears to contain an environment variable, such as %ServerName%, Jukebox Studio will attempt to resolve the environment variable on the Cloudifying PC. If there is no such environment variable, Jukebox Studio will conclude that the variable name is a constant and will escape the "%" characters. %ServerName% will become %%ServerName%%.

In order to preserve the environment variable in the cloudified application, define the environment variable on the Cloudifying PC before creating the appset.

Editing Registry Values

You can edit registry values after they have been added.

To edit a registry value:

1. Right-click the value you wish to edit.
2. Select **Edit Value** from the pop-up menu. The **Edit Registry Value** dialog appears.

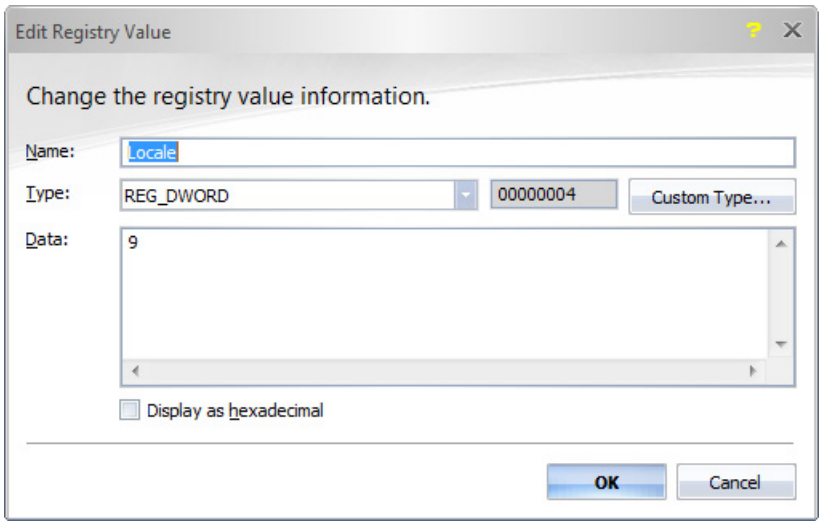


Figure 7-13: Edit Registry Value Dialog

3. Make the necessary changes. For more information, see Adding Registry Values on page 87.
4. Click **OK**. The Registry view will be updated to reflect your changes.

Deleting Registry Keys and Values

It is possible to delete any registry key or value that was created during installation or that was added manually.

To delete a registry key or value:

1. Right-click the registry key or value that you wish to delete.
2. Select **Delete** from the pop-up menu.
3. Select **OK** when prompted for confirmation.

Excluding Registry Keys

Excluded registry keys are not included when the appset is cloudified, but are still part of the project. This gives you the flexibility of including certain registry keys for one deployment of an appset and not for another. Excluding registry keys also enables you to test an appset without a registry key before permanently deleting it from the project.

To exclude a Registry Key:

1. Right-click a registry key that you wish to exclude.
2. Select **Exclude** from the pop-up menu.

3. Excluded registry keys are shown with their name and properties in italicized text.
4. If you wish to exclude another registry key, repeat steps 1-3.

Modifying Environment Variables

The Environment view is accessed by selecting **Environment** from the **View** menu, or by selecting the **Environment** view as shown below.

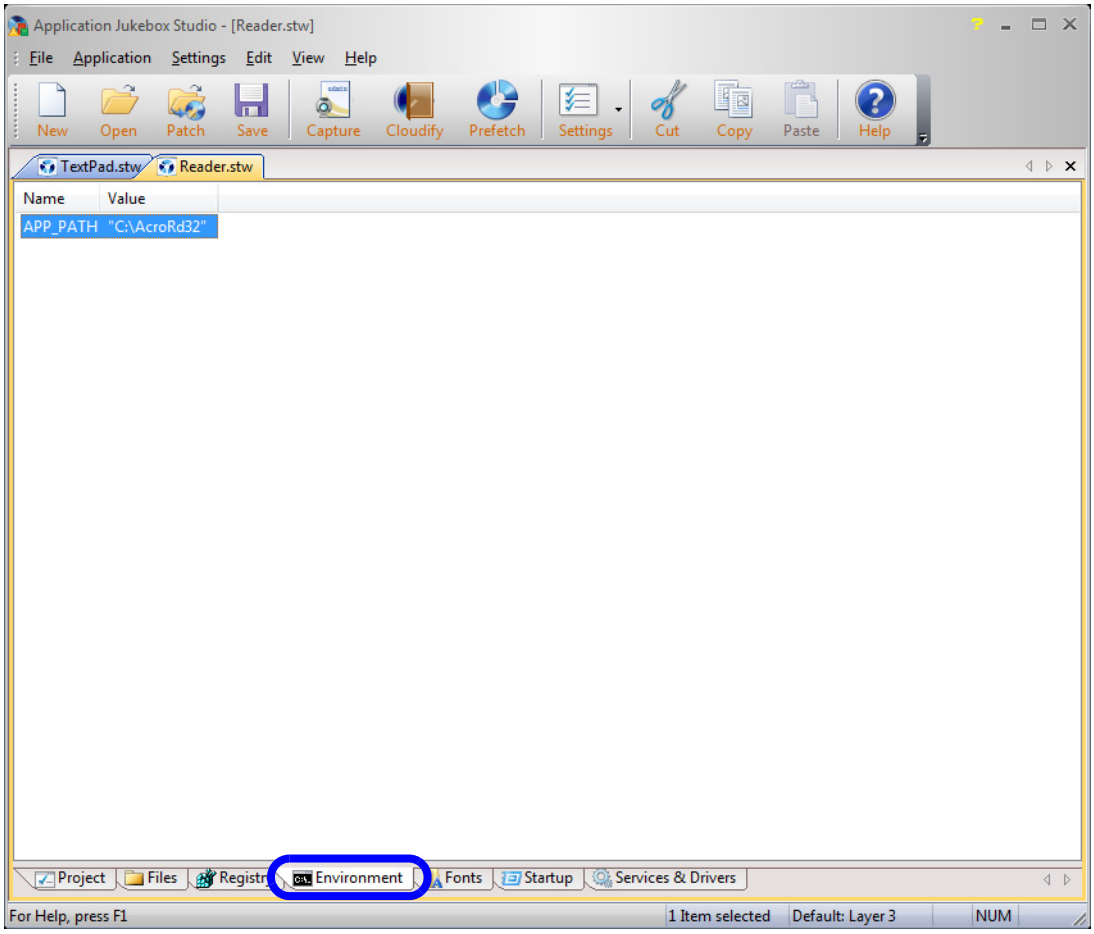

Application Jukebox Studio - [Reader.stw]
File Application Settings Edit View Help
New Open Patch Save Capture Cloudify Prefetch Settings Cut Copy Paste Help
TextPad.stw Reader.stw
Name Value
APP_PATH "C:\AcroRd32"
Project Files Registry Environment Fonts Startup Services & Drivers
For Help, press F1
1 Item selected
Default: Layer 3
NUM

Figure 7-14: Environment view

This view displays environment variables that have been created either during installation or manually added. Most relevant commands can be accessed either from the **Edit** menu or by right-clicking on an environment variable.

There are two types of environment variables: user environment variables (specific for each user and set in the registry under HKEY_CURRENT_USER \ Environment) and system environment variables (global for everyone and set in the registry under HKEY_LOCAL_MACHINE \ SYSTEM \ CurrentControlSet \ Control \ Session Manager \ Environment). When an application installation is captured, Jukebox Studio will record the changes to the environment variables from both the user and system locations and merged the results into the Environment view. These environment variables are then injected into either the application or to the system depending on the **Environment Variables** setting under the **Virtualization** screen on the **Settings** dialog. See Virtualization Settings on page 62.

- Environment variables shared with the entire system will be added to the HKEY_CURRENT_USER \ Environment registry key when the appset is virtualized making them available to all applications for that specific user.
- Environment variables visible only to the application are set in the process environment for the application when it is launch from Jukebox Player. Child-processes of the application may inherent these environment variables but no other process on the system can access them.

All virtual environment variables will override any existing variables of the same name for either method. The only exception to this rule is for the %PATH% or %PATHEXT% variables. These will automatically be appended to exists system values much like:

PATH = %PATH%;c:\newfolder

Visit **http://support.microsoft.com/kb/104011** for more general information on setting environment variables.

NOTE

Some legacy applications may check the registry for environment variables under a specific hive and not use the variables in the process environment. For such applications, it may be necessary to add the environment variables directly into the Registry view in the correct locations and set the key disposition to Layer 4.

Adding Environment Variables

You can create new environment variables as described below.

To add an environment variable:

1. Right-click a blank area of the Environment view.
2. Select **Add Environment Variable** from the pop-up menu.
   The **Add Environment Variable** dialog appears.

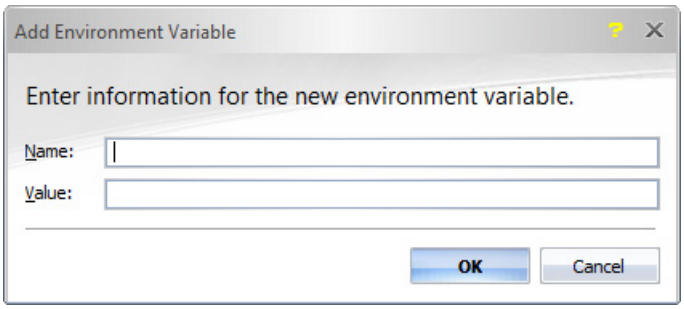


Figure 7-15: Add Environment Variable Dialog

3. Enter a name for the new environment variable in the **Name** field.
4. Enter a value for the variable in the **Value** field.
5. Click **OK**. The Environment view will be updated to reflect the added variable.

Editing Environment Variables

To edit a System environment variable:

1. From the Environment view, right-click the environment variable that you wish to edit.
2. Select **Edit Environment Variable** from the pop-up menu.
   The **Edit Environment Variable** dialog appears.

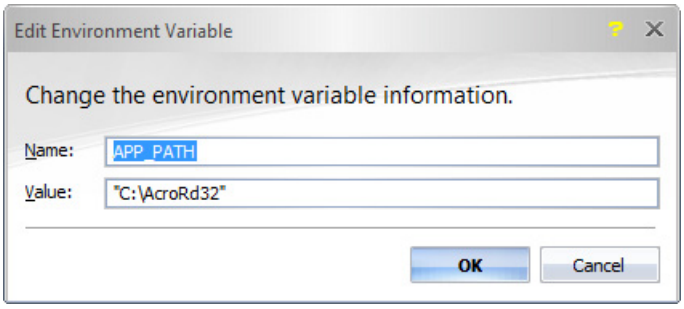


Figure 7-16: Edit Environment Variable Dialog

3. Make the necessary changes. For more information, see Adding Environment Variables on page 90.
4. Click **OK**. The Environment view will be updated to reflect your changes.

Deleting Environment Variables

It is possible to delete any environment variable that was created during installation or that was added manually.

To delete an environment variable:

1. Right-click the environment variable that you wish to delete.
2. Select **Delete** from the pop-up menu.
3. Select **OK** when prompted for confirmation.

Excluding Environment Variables

Excluded environment are not included when the appset is cloudified, but are still part of the project. This gives you the flexibility of including certain environment variables for one deployment of an appset and not for another. Excluding environment variables also enables you to test an appset without an environment variable before permanently deleting it from the project.

To exclude a Environment Variable:

1. Right-click an environment variable that you wish to exclude.
2. Select **Exclude** from the pop-up menu.
3. Excluded environment variables are shown with their name and properties in italicized text.
4. If you wish to exclude another environment variable, repeat steps 1-3.

Modifying Fonts

Fonts installed during the capture process will be shown on the Fonts view. You can manually add other fonts to a project. Most relevant commands can be accessed either from the **Edit** menu or by right-clicking on a font asset.

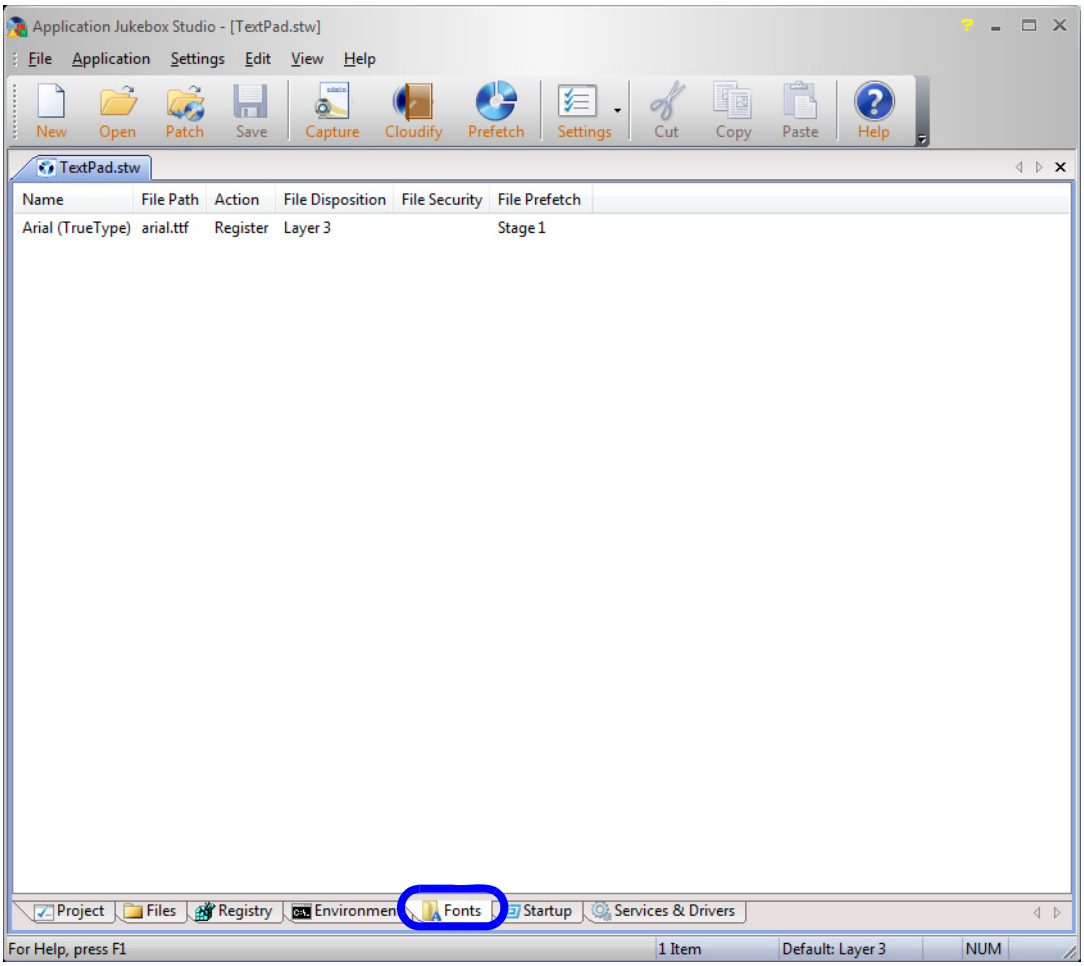

Application Jukebox Studio - [TextPad.stw]
File Application Settings Edit View Help
New Open Patch Save Capture Cloudify Prefetch Settings Cut Copy Paste Help
TextPad.stw
Name File Path Action File Disposition File Security File Prefetch
Arial (TrueType) arial.ttf Register Layer 3 Stage 1
Project Files Registry Environmen Fonts Startup Services & Drivers
For Help, press F1
1 Item
Default: Layer 3
NUM

Figure 7-17: The Fonts View

The fonts in the appset will either be available only to the appset (injected) or made available to other applications on the Target PC (registered), according to the Font setting under the Virtualization screen on the **Settings** dialog. See Virtualization Settings on page 62.

NOTE

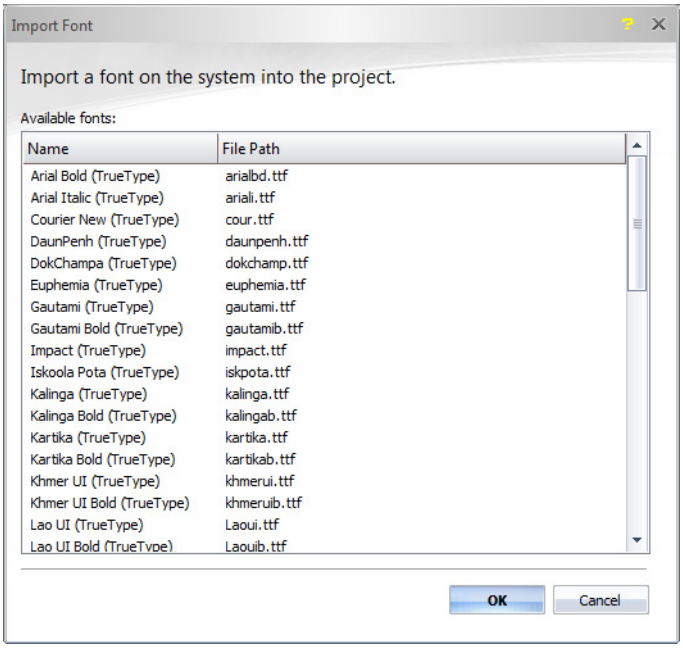
To add more fonts to a project, you will need to know the file name or names for the font, and not just the font name. Keep in mind that in many cases, there is more than one file for each font. For example, “Arial” consists of four files, one for the basic font, another for italic, bold, and bold italic.

Adding Fonts

There are two ways to add a font to the project: Importing system fonts or selecting the font files directly. This second option is useful if adding fonts that are not installed on the Cloudifying PC.

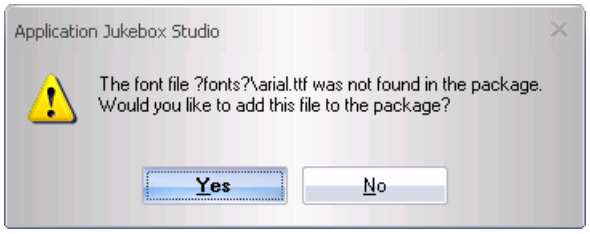
To import a system font into a project:

1. Right-click a blank area of the Fonts view.
2. Select **Import Font** from the pop-up menu. The **Import Font** dialog appears.



3. Select the desired system font.
4. Click **OK**.

   If the corresponding font file is found on the Cloudifying PC, Jukebox Studio will prompt you, asking if you want to add the file to the package.


Application Jukebox Studio
The font file ?fonts?\arial.ttf was not found in the package.
Would you like to add this file to the package?
Yes
No

   This may not be necessary if you are confident the font will be present on the Target PC.

   If the selected file is not found on the Cloudifying PC, then Jukebox Studio will alert you, asking if you still wish to add the font to project.

   If the font file is not added to the project, the file path will be shown in red in the Fonts view.

5. The Fonts view will be updated to reflect the added font.

To manually add a font to a project:

1. Right-click a blank area of the Fonts view.
2. Select **Add Font** from the pop-up menu. The **Add Font** dialog appears.

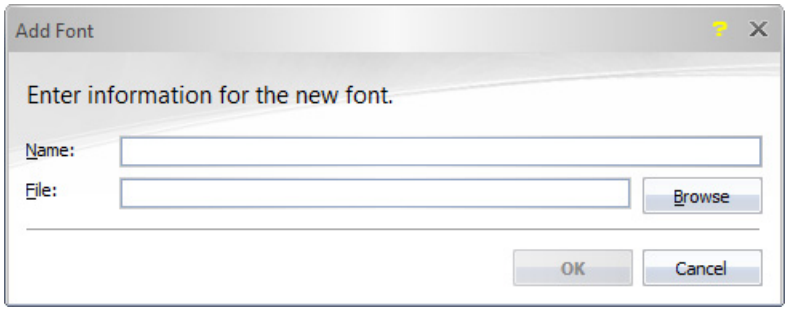
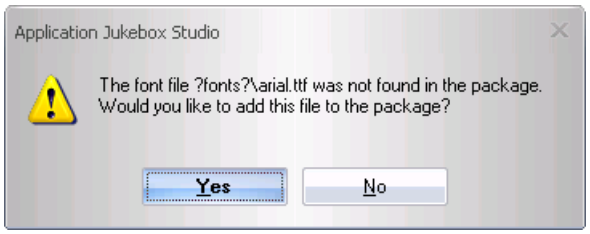


Figure 7-18: The Add Font dialog

3. Enter a name for the new font in the **Name** field.
4. Click **Browse** and locate the font file in the **File** field.
5. Click **OK**.

   If the corresponding font file is found on the Cloudifying PC, Jukebox Studio will prompt you, asking if you want to add the file to the project.


Application Jukebox Studio
The font file ?fonts?\arial.ttf was not found in the package.
Would you like to add this file to the package?
Yes
No

   This may not be necessary if you are confident the font will be present on the Target PC.

   If the selected file is not found on the Cloudifying PC, then Jukebox Studio will alert you, asking if you still wish to add the font to project.

   If the font file is not added to the project, the file path will be shown in red in the Fonts view.
6. The Fonts view will be updated to reflect the added font.

NOTE

If the virtualization setting for fonts is set to register, but the disposition for the corresponding font file is set to virtual-isolated (layer 4), the disposition for the font will be shown in red. This font will not be visible to other applications on the Target PC, despite the virtualization setting. For more information about conflict alerts, see Conflict Color Coding on page 115.

Editing Fonts

Existing fonts can be edited, their name or the file changed if necessary.

To edit a font:

1. From the Fonts view, right-click the font that you wish to edit.
2. Select **Edit Font** from the pop-up menu. The **Edit Font** dialog appears.

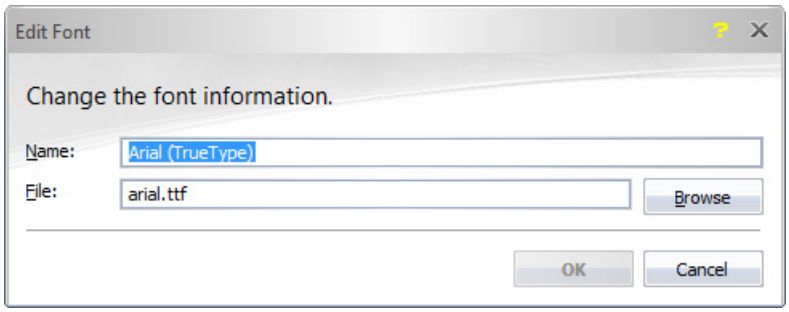


Figure 7-19: The Edit Font Dialog

3. Make the necessary changes. For more information, see Adding Fonts on page 93.
4. Click **OK**. The Fonts view will be updated to reflect your changes.

Editing the File Properties of a Font

Adding a Font asset also adds the file to the Files view. You can edit the prefetch, disposition, and security settings from within the Fonts view, rather than having to locate and edit the file from the Files view.

To edit the file properties of a Font asset:

1. Right-click the asset with properties you wish to edit.
2. Select **Edit File Properties** from the pop-up menu. The **Font File Properties** dialog appears.

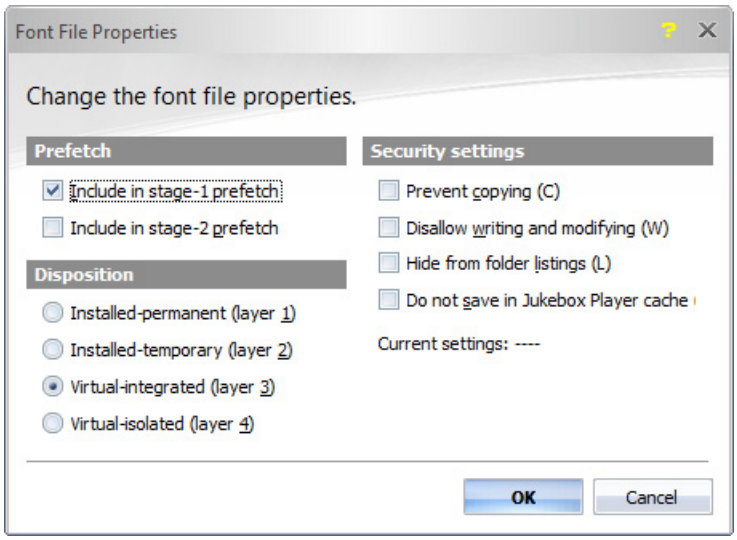


Figure 7-20: Fonts View, Font File Properties Dialog

3. Make the necessary changes. For more information, see Editing File Properties on page 80.

4. Click **OK**. The Fonts view will be updated to reflect the changes.

Deleting Fonts

It is possible to delete any font that was created during installation or that was added manually.

To delete an font:

1. Right-click the font that you wish to delete.
2. Select **Delete** from the pop-up menu.
3. Select **OK** when prompted for confirmation.

Excluding Fonts

Excluded fonts are not included when the appset is cloudified, but are still part of the project. This gives you the flexibility of including certain fonts for one deployment of an appset and not for another. Excluding fonts also enables you to test an appset without a font before permanently deleting it from the project.

To exclude a Font:

1. Right-click a font that you wish to exclude.
2. Select **Exclude** from the pop-up menu.
3. Excluded fonts are shown with their name and properties in italicized text.
4. If you wish to exclude another font, repeat steps 1-3.

Locating the File or Registry Entry of a Font

If you need to view the Files View entry or the registry key for the selected font, Jukebox Studio will take you to the correct asset directly from the Fonts View.

Locating the font file in the Files View

1. Right-click the font for which you wish to view the command file.
2. Click **Jump To** and select **Font File**.
3. Jukebox Studio will switch to the Files View, select the correct folder and highlight the appropriate file.

Locating the registry value in the Registry View

1. Right-click the font for which you wish to view the registry value.
2. Click **Jump To** and select **Fonts Value**.
3. Jukebox Studio will switch to the Registry View, select the correct key, and highlight the value.

Modifying Startup Items

Startup items are programs that typically launch at system startup, which are installed along with the paged application. These often include the system tray icons that permit direct access to the functions of the paged application, even when the paged application is not running.

Any startup items created during installation will be visible in the Startup view. Others can be added, and any item can be deleted or excluded.

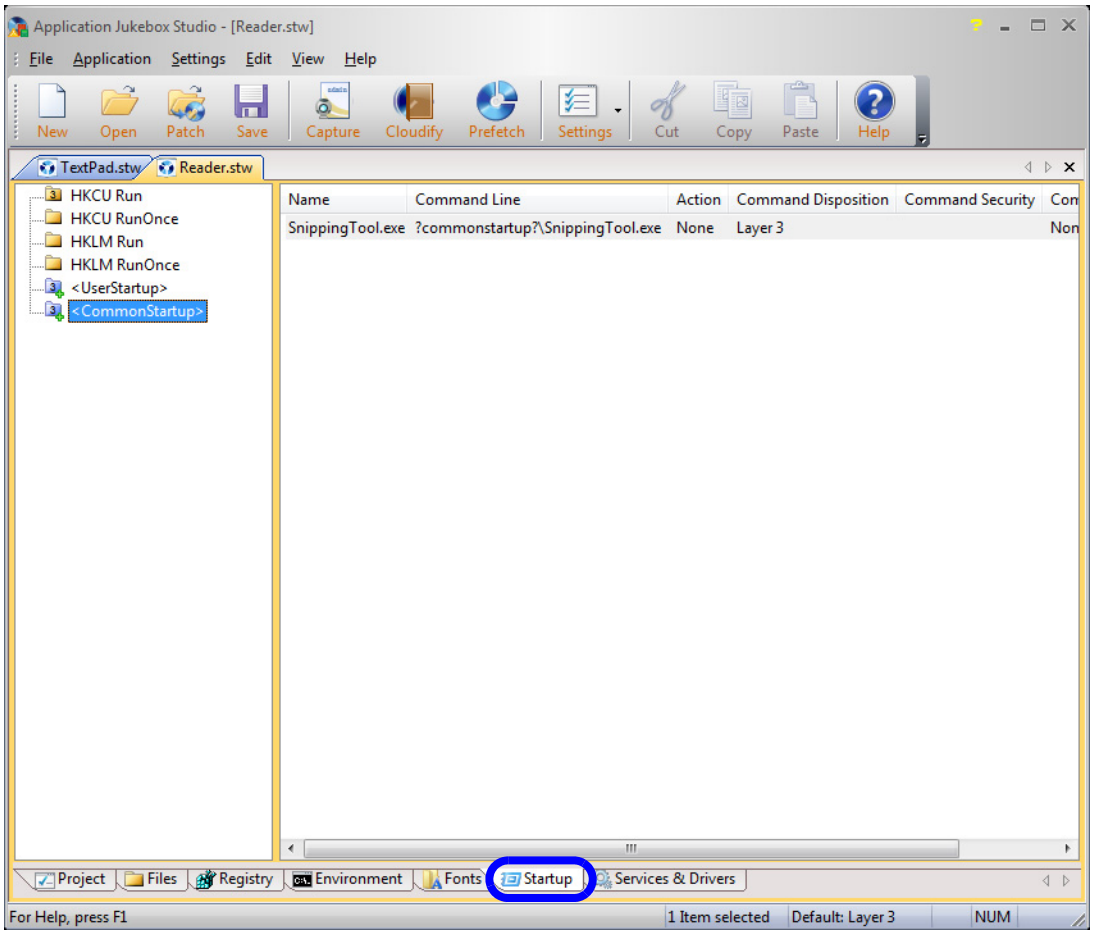

Application Jukebox Studio - [Reader.stw]
File Application Settings Edit View Help
New Open Patch Save Capture Cloudify Prefetch Settings Cut Copy Paste Help
TextPad.stw Reader.stw
HKCU Run
HKCU RunOnce
HKLM Run
HKLM RunOnce
<UserStartup>
<CommonStartup>
Name Command Line Action Command Disposition Command Security
SnippingTool.exe ?commonstartup?\SnippingTool.exe None Layer 3
Project Files Registry Environment Fonts Startup Services & Drivers
For Help, press F1
1 Item selected
Default: Layer 3
NUM

Figure 7-21: Startup View

Types of Startup Items

Windows launches Startup items using one of two different methods: Registry "Run" Key items and Start Menu items. Both kinds of captured startup items can be either enabled or disabled via the **Virtualization** screen on the **Settings** dialog. See Virtualization Settings on page 62.

Registry Run Key Items

The path to the executable is found in the value of special registry keys called *Run Keys*. These startup items are found in one of the yellow folders on the **Startup** view:

- **HKCU Run** ("HKCU" is short for "HKEY_CURRENT_USER") – These startup items apply only to the currently logged-in user.
- **HKCU RunOnce** – Similar to HKCU Run items, but these startup items will only run once. If one of these items has already been loaded before the cloudified application is virtualized on the Target PC, it will not load again.
- **HKLM Run** ("HKLM" is short for "HKEY_LOCAL_MACHINE") – These startup items apply to all local users on the Target PC.

- **HKLM RunOnce** – Similar to the “HKCU RunOnce” startup items in that they will only load once and not every time the cloudified application is virtualized on the Target PC. Like the “HKLM Run” startup items, these apply to all local users on the Target PC.

NOTE

Adding a startup item to the registry does not add the corresponding executable file to the project. This may not be necessary if you are confident that the executable will be present on the Target PC. If the corresponding file is not present in the File view, the path will be shown in red in the Startup View. For more information about conflict alerts, see Conflict Color Coding on page 115.

Start Menu Items

These items appear as shortcuts in the **Startup** submenu under the Programs Start Menu. These items are found in the two blue folders on the **Startup** view:

- **UserStartup** – These Startup items apply only to the currently logged-in user.
- **CommonStartup** – These Startup apply to all local users on the Target PC.

NOTE

Adding a startup item to the start menu does not add the corresponding executable file to the project. This may not be necessary if you are confident that the executable will be present on the Target PC. If the corresponding file is not present in the File view, the path will be shown in red in the Startup View. For more information about conflict alerts, see Conflict Color Coding on page 115.

Adding Startup Items

Before adding a Startup Item, you need to have the following information:

- Registry or Start Menu
- Name and location of the registry key (if necessary).
- Location and name of the executable file.

There are two different methods, depending on the type of Startup item being added.

To add a registry key Startup item:

1. Right-click either the HKCU or the HKLM folder, or select one of these folders and right-click in a blank area within the item view.
2. Select **Add Startup Item...** from the pop-menu. The **Add Startup Value** dialog appears.

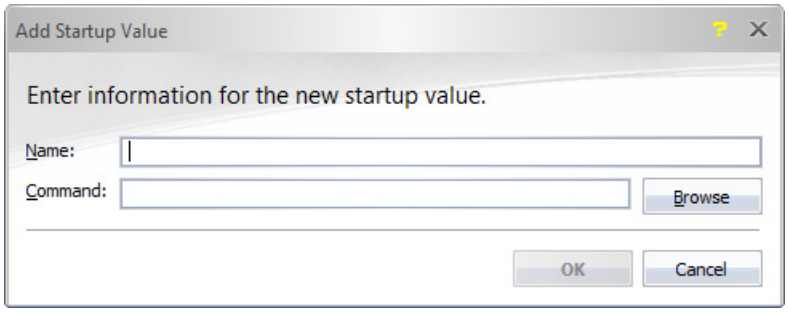


Figure 7-22: Startup View Add Startup Value dialog

3. Enter the **Name** of the new Registry Key.
4. In **Command**, enter the full path to the executable file.
5. Click **OK**.

Adding Start Menu Startup Items

1. Right-click either the UserStartup or the CommonStart folder, or select one of these folders and right-click in a blank area within the item view.
2. Select **Add Startup Item...** from the pop-menu. The **Add File** dialog appears.

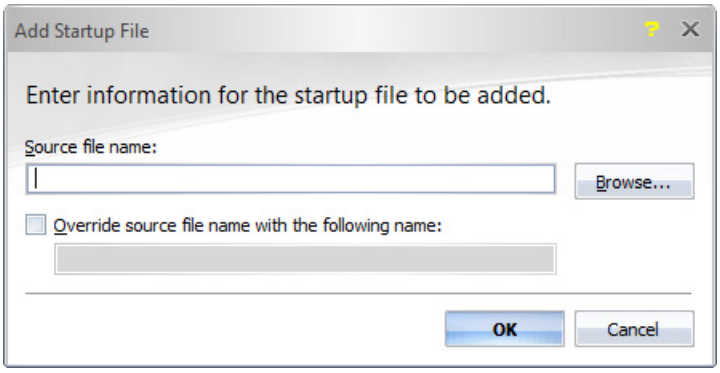


Figure 7-23: Startup View Add Startup File dialog

3. Enter the path of the file you wish to add or browse to it.
4. Normally, the selected file is added with its current name. If you wish to add the file under a different name, select **override source file name with the following name**.
5. Enter the new file name.
6. Click **OK**. The Startup view will be updated to reflect the added file.

Editing Startup Items

Editing Startup items can also be done from the File or Registry views, depending on the type, but it is usually simpler to edit them directly from the Startup view.

Editing Registry Key Startup Items

1. Right-click the item with properties you wish to edit.
2. Select **Edit** from the pop-up menu. The **Edit Startup Value** dialog appears.

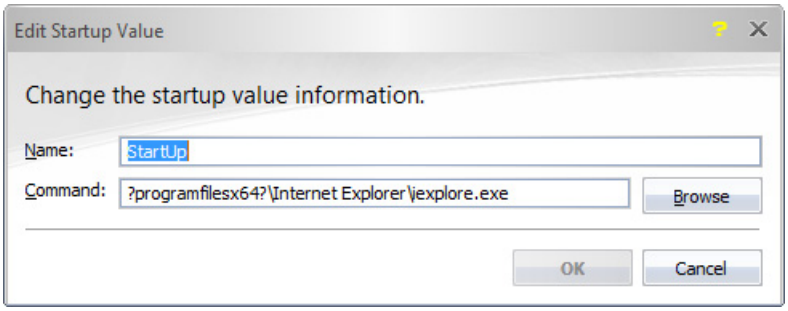


Figure 7-24: Startup View Edit Startup Value dialog

3. Make the necessary changes. For more information, see To add a registry key Startup item: on page 98.

4. Click **OK**. The Startup view will be updated to reflect the changes.

Editing Startup Menu Items

1. Right-click the item with properties you wish to edit.
2. Select **Edit** from the pop-up menu. The **Edit File** dialog appears.
3. Make the necessary changes. For more information, see Adding Start Menu Startup Items on page 99.
4. Click **OK**. The Startup view will be updated to reflect the changes.

NOTE

If the disposition of the registry key for a startup item is installed-permanent (layer 1) or installed-temporary (layer 2), but the corresponding executable file is set to virtual-integrated (layer 3) or virtual-isolated (layer 4), the item will be flagged with the disposition in red, because the registry key will be physically present on the host PC, while the file to which it points is not. For more information about conflict alerts, see Conflict Color Coding on page 115.

Editing the Command Line File Properties of a Start Menu Item

Adding a Start Menu Startup item also adds the file to the Files view. You can edit the prefetch, disposition, and security settings from within the Startup view, rather than having to locate and edit the file from the Files view.

To edit the file properties of a Startup item:

1. Right-click the item with properties you wish to edit.
2. Select **Edit Command Line File Properties** from the pop-up menu. The **Edit Command Line Properties** dialog appears.

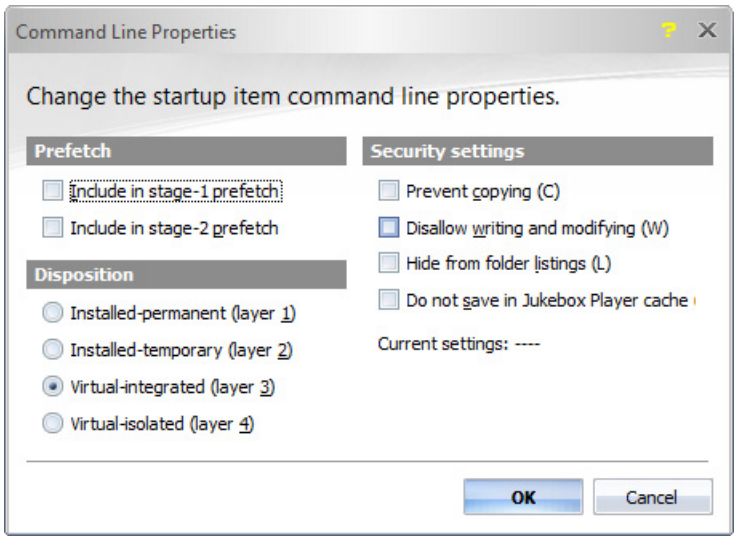


Figure 7-25: Startup View, Edit File Properties Dialog

3. Make the necessary changes. For more information, see Editing File Properties on page 80.
4. Click **OK**. The Startup view will be updated to reflect the changes.

NOTE

If a startup item points to a shortcut (.lnk file) which has a disposition of installed-permanent (layer 1) or installed-temporary (layer 2), but the target executable file is set to virtual-integrated (layer 3) or virtual-isolated (layer 4), the item will be flagged with the disposition in red, because the shortcut will be physically present on the host PC, while the file to which it points is not. For more information about conflict alerts, see Conflict Color Coding on page 115.

Deleting Startup Items

It is possible to delete any Startup item that was created during installation or that was added manually.

To delete a Startup item:

1. Right-click the Startup item that you wish to delete.
2. Select **Delete** from the pop-up menu.
3. Select **OK** when prompted for confirmation.

Excluding Startup Items

Excluded Startup items are not included when the appset is cloudified, but are still part of the project. This gives you the flexibility of including certain Startup items for one deployment of an appset and not for another. Excluding items also enables you to test an appset without a Startup item before permanently deleting it from the project.

To Exclude a Startup item:

1. Right-click the Startup item that you wish to exclude.
2. Select **Exclude** from the pop-up menu.
3. Excluded items are shown with their name and properties in italicized text.
4. If you wish to exclude another item, repeat steps 1-3.

Locating the File or Registry Entry of a Startup Item

If you need to view the Files View entry for a command file or the registry key for the selected startup item, Jukebox Studio will take you to the correct item directly from the Startup View.

Locating the command file in the Files View

1. Right-click the startup item for which you wish to view the command file.
2. Click **Jump To** and select **Startup Command File**.
3. Jukebox Studio will switch to the Files View, select the correct folder and highlight the appropriate file.

Locating the registry value in the Registry View

1. Right-click the startup item for which you wish to view the registry value.
2. Click **Jump To** and select **Startup Value**.
3. Jukebox Studio will switch to the Registry View, select the correct key, and highlight the value.

Modifying Services and Drivers

Services installed during the capture process will be shown on the Services & Drivers view. You can manually add other services to a project. Most relevant commands can be accessed either from the **Edit** menu or by right-clicking on a service item.

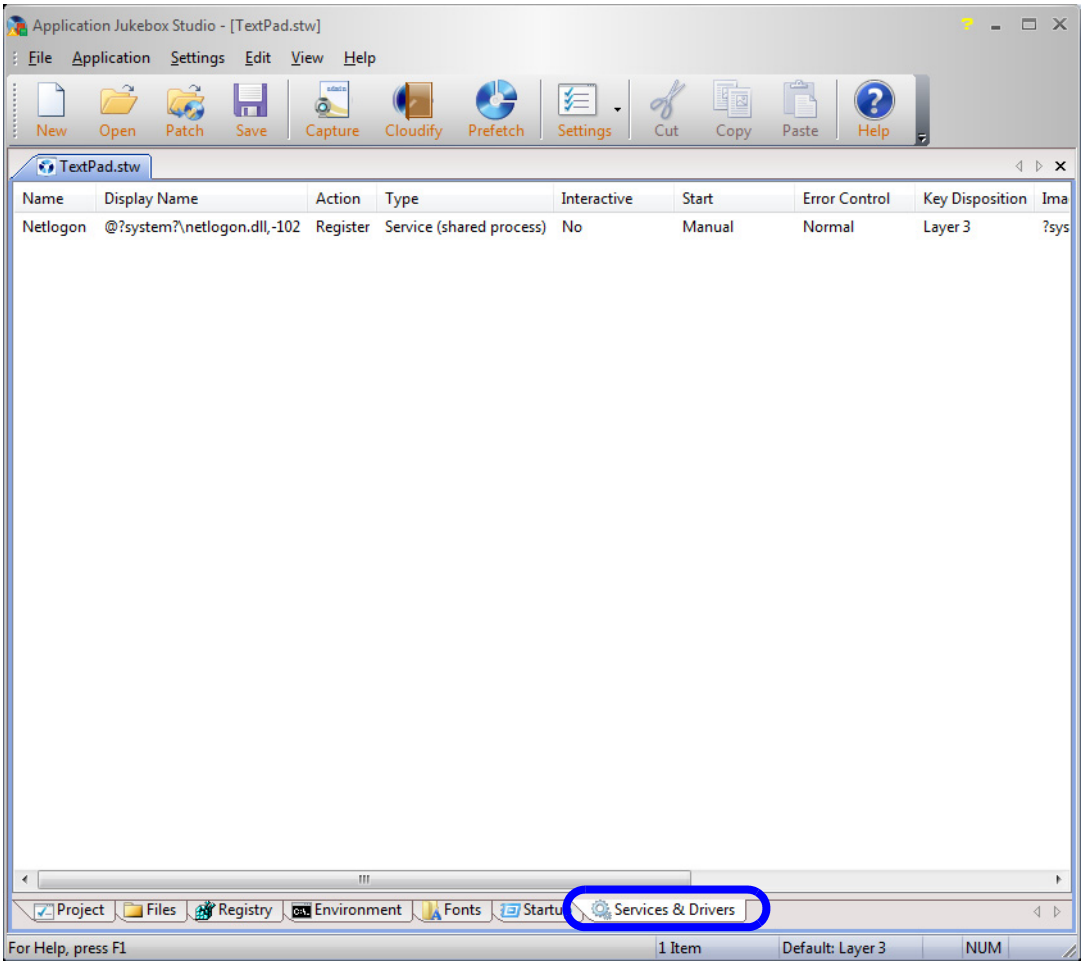

Application Jukebox Studio - [TextPad.stw]
File Application Settings Edit View Help
New Open Patch Save Capture Cloudify Prefetch Settings Cut Copy Paste Help
TextPad.stw
Name Display Name Action Type Interactive Start Error Control Key Disposition Ima
Netlogon @?system?\netlogon.dll,-102 Register Service (shared process) No Manual Normal Layer 3 ?sys
Project Files Registry Environment Fonts Startu Services & Drivers
For Help, press F1 1 Item Default: Layer 3 NUM

Figure 7-26: The Services & Drivers view

NOTE

Most of what follows applies to both services and drivers since a driver is a special kind of service. So in the remainder of this section we will not distinguish between the two unless the described functionality or behavior is specific to one or the other.

Driver and services are shared components within Windows, which means that while the files and registry keys can be virtual, the running process will affect the entire system. Certain types of services or drivers must be ran at system startup and these, as well as any dependencies, may need to be made to layer-2. Removing such a service or driver may then require a system reboot by using a CAE.

Adding Services

To add a service to a project:

1. Right-click a blank area of the Services & Drivers view.
2. Select **Import Service** from the pop-up menu. The **Import Service** dialog appears.

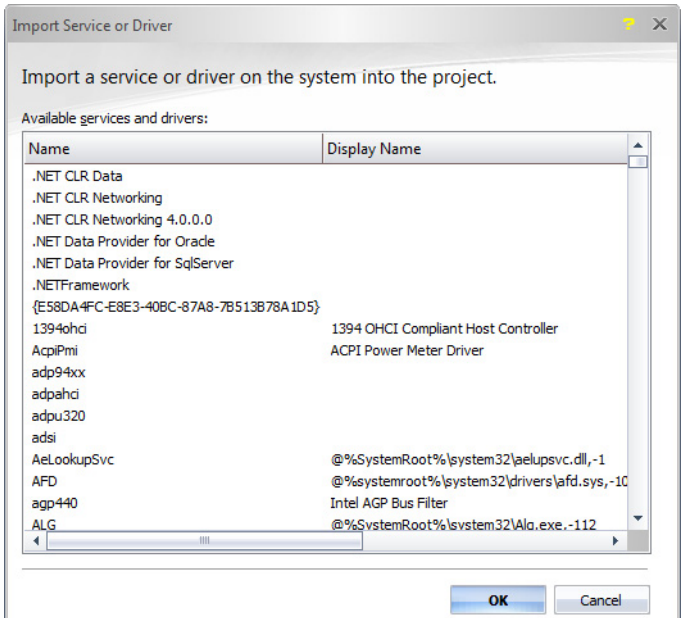
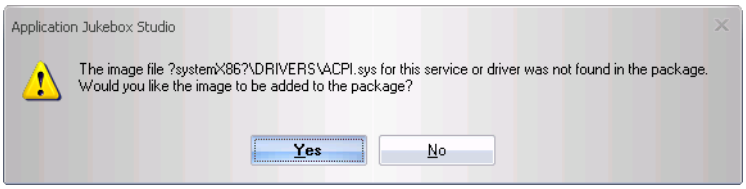


Figure 7-27: The Import Service dialog

3. Select the service from the list.
4. Click **OK**.
5. Jukebox Studio will prompt you to add the corresponding image file, if necessary.


Application Jukebox Studio
The image file ?system%86?\DRIVERS\ACPI.sys for this service or driver was not found in the package.
Would you like the image to be added to the package?
Yes
No

If you are certain the service or driver image file will be present on the Target PC when the application is virtualize, you can click **No**. Click **Yes** to add the image file to the project. It will appear in the Files view.

6. The Service & Drivers view will be updated to reflect the added service.

Editing the Registry Key Properties of a Service

To change the disposition of a registry key:

1. Right-click the desired item, and select **Service Key Disposition** from the pop-up menu.

   The **Service Key Disposition** dialog appears.

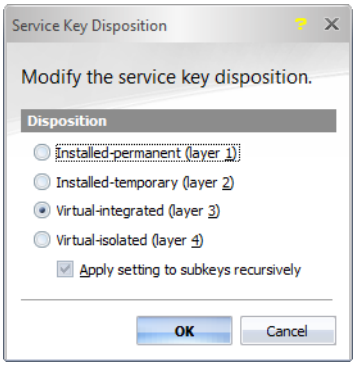


Figure 7-28: Service Key Disposition Dialog

2. Select the disposition. For more information about disposition settings, see Changing the Disposition of a Registry Key on page 86.
3. The **Apply setting to subkeys recursively** setting will be checked automatically.
4. Click **OK**.

The Services & Drivers view will be updated to reflect the changed disposition setting for the selected item.

Editing the Image File Properties of a Service

If the corresponding file for a Services & Drivers item has also been added to the Files view, you can edit the prefetch, disposition, and security settings from within the Services view, rather than having to locate and edit the file from the Files view.

NOTE

Any services in an appset on layer 3 are guaranteed to have all of their dependencies (e.g. .dll, .ini, .manifest, .cfg, etc. files) available to the when they need them. When setting services assets to physical layers (layer 1 or 2) however, you must be sure any of these dependencies are also layer 1 or 2. This is important if the services are meant to automatically start at Windows logon. Dependency Walker is a tool that helps identifying DLL files that an executable needs to run successfully.

To edit the file properties of a service:

1. Right-click the item with properties you wish to edit.
2. Select **Edit Image File Properties** from the pop-up menu. The **Image File Properties** dialog appears.

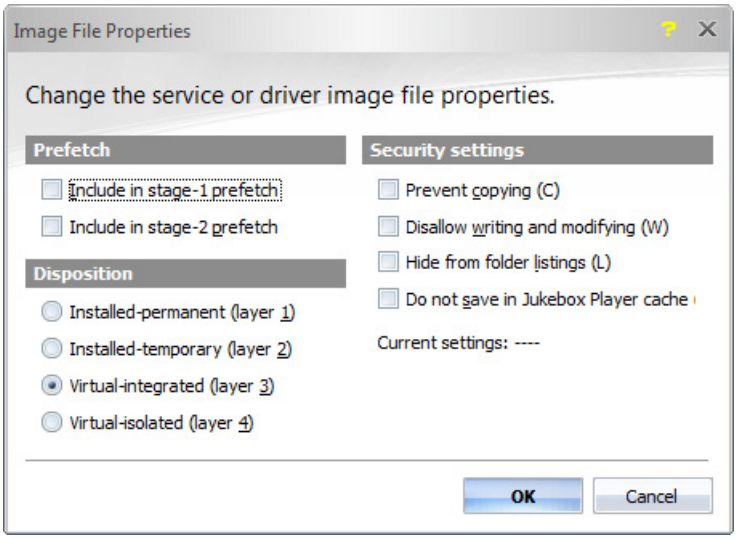


Figure 7-29: Service & Drivers View, Image File Properties Dialog

3. Make the necessary changes. For more information, see Editing File Properties on page 80.
4. Click **OK**. The Startup view will be updated to reflect the changes.

NOTE

If there is a conflict involving one of the items, the relevant fields will appear in red. For details about the specific reason for this alert, see Conflict Color Coding on page 115.

Editing Services

It is possible to directly edit the operating system properties dialog for any service that was created during installation or that was added manually. These are also accessible by opening Control Panel > Administrative Tools > Services, right-clicking on the name of a service, and selecting properties. Jukebox Studio is providing a shortcut to the properties dialog from within the application.

To edit a service:

1. Right-click the service that you wish to edit.
2. Select **Edit** from the pop-up menu.

3. Windows will display the properties dialog for that service.

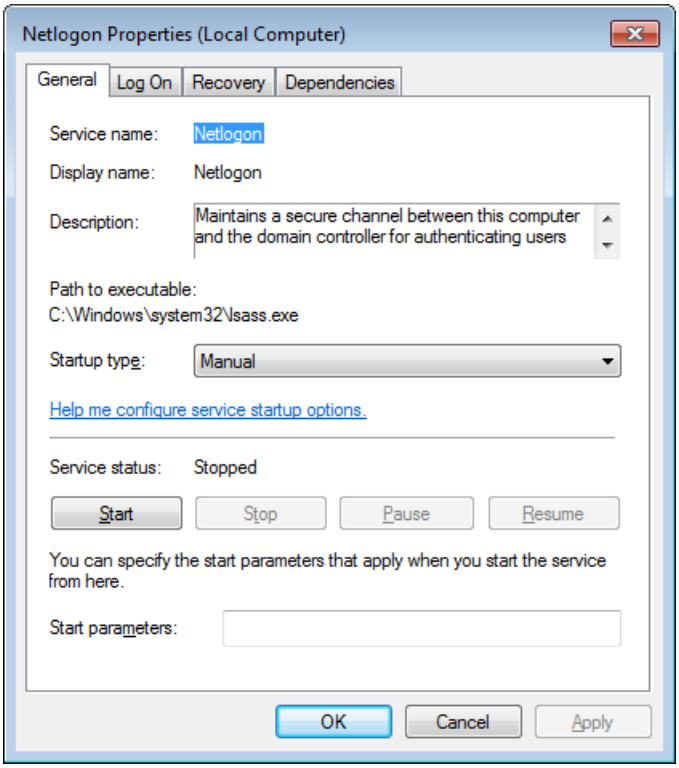


Figure 7-30: Sample Windows service properties dialog

4. Select **OK** when finished to close the dialog, or **Cancel** if no changes are made.

To edit a driver:

1. Right-click the driver that you wish to edit.
2. Select **Edit** from the pop-up menu.
3. A dialog will display the properties for that driver
4. Select **boot loader**, **system**, **automatic**, **manual**, or **disabled** for the driver..

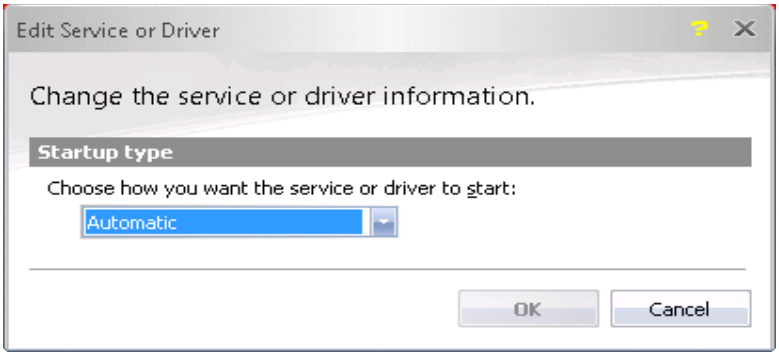


Figure 7-31: Sample Windows service properties dialog

5. Select **OK** when finished to close the dialog, or **Cancel** if no changes are made.

Setting Virtualization Actions for a Service

The virtualization action determines how the selected service or driver is handled upon virtualization of the application on the Target PC.

To edit the virtualization actions for a service:

1. Right-click the service that you wish to edit.
2. Select **Change Virtualization Action** from the pop-up menu.
3. Jukebox Studio will display the **Change Service or Driver Virtualization Action** dialog for that service.

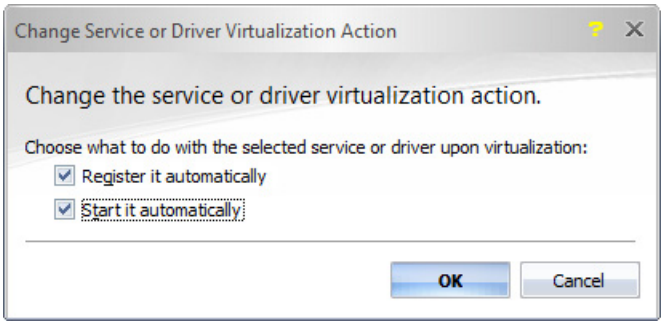

Change Service or Driver Virtualization Action
Change the service or driver virtualization action.
Choose what to do with the selected service or driver upon virtualization:
Register it automatically
Start it automatically
OK
Cancel

Figure 7-32: Service & Drivers View, Change Service or Driver Virtualization Action Dialog

4. Check the desired settings. There are two options:

   Register them automatically — Any services and drivers installed with the application will be registered with the Target PC upon virtualization.

   Start them automatically – Registered services and drivers are started upon virtualization.
5. Select **OK** when finished to close the dialog, or **Cancel** if no changes are made.

Changing the Failure Action of a Service

Windows services will attempt to restart themselves if they crash for any reason. Jukebox Studio allows to you to disable this for some services that are part of an appset.

To change the failure action for a service:

1. Right-click the service for which you wish to change the failure action.
2. Select **Turn Failure Actions OFF** from the pop-up menu, if restart is enabled, or **Turn Failure Actions ON** from the pop-up menu, if restart is already disabled.
3. The option will be grayed out if the failure action cannot be changed for the selected service or driver.
4. If failure actions are ON for a service or driver, the name of that service will be shown in red on the Services & Drivers view.

Deleting Services

It is possible to delete any service that was created during installation or that was added manually.

To delete a service:

1. Right-click the service that you wish to delete.
2. Select **Delete** from the pop-up menu.

3. Select **OK** when prompted for confirmation.

Excluding Services

Excluded services are not included when the appset is cloudified, but are still part of the project. This gives you the flexibility of including certain services for one deployment of an appset and not for another. Excluding services also enables you to test an appset without a services before permanently deleting it from the project.

To exclude a service:

1. Right-click the service that you wish to exclude.
2. Select **Exclude** from the pop-up menu.
3. Excluded services are shown with their name and properties in italicized text.
4. If you wish to exclude another services, repeat steps 1-3.

Locating the Image File or Registry Key of a Service

If you need to view the Files View entry for an Image File or the Registry Key for the selected service or driver, Jukebox Studio will take you to the correct asset directly from the Services & Drivers View.

Locating the image file in the Files View

1. Right-click the service for which you wish to view the image file.
2. Click **Jump To** and select **Service File**.
3. Jukebox Studio will switch to the Files View, select the correct folder and highlight the appropriate file.

Locating the registry key in the Registry View

1. Right-click the service for which you wish to view the registry key.
2. Click **Jump To** and select **Service Key**.
3. Jukebox Studio will switch to the Registry View and select the correct key.

Searching for Assets

Finding Assets

The **Find** feature can be used to locate folders, files, registry keys and values that match specific criteria.

To find assets:

1. From any view, select the **Find** menu under the **View** menu (or press F3). The **Find** dialog appears.

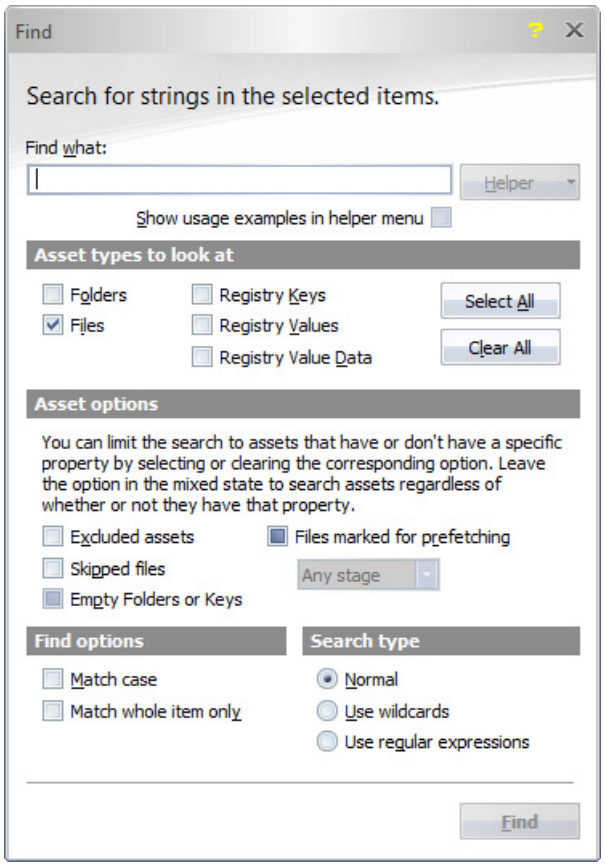


Figure 7-33: Find Dialog

2. Select the type of search: **Normal**, **Using wildcards**, or **Using regular expressions** (see About Search Types on page 111).
3. Choose the areas to search under **Look at**:
   - **Folders** - Names of the directories seen in the File view.
   - **Files** - Names of the files found in folders seen in the File view.
   - **Registry Keys** - Names of the registry keys seen in the Registry view.
   - **Registry Values** - Strings contained within the registry keys seen in the Registry view.
   - **Registry Value Data** – The strings or digits assigned to Registry values.
4. Check or uncheck the desired **Asset Options**:

   These check boxes start up in a mixed state, neither checked nor empty, meaning that search will not consider the selected option or options. Not all options are available for all combinations of asset

types. For example, skipped files applies to files and files only. If you select any asset type other than files, even if you also select files, skipped files will not be available during that search.

- **Excluded assets** – Checking this option will restrict the search to assets marked as Excluded in any view. Unchecking this option will restrict the search to assets not marked as Excluded.
- **Skipped files**— These files were missing or inaccessible during packaging. These files are also automatically marked as Excluded. Checking this option will restrict the search to skipped files. Unchecking this option will restrict the search to files that were not skipped.
- **Empty Folders or Keys** – Checking this option will restrict the search to folders that contain no files or registry keys with no values. Unchecking this option will restrict the search to non-empty assets.
- **Files marked for prefetching** – Check this option will restrict the search to files marked for prefetching during the selected stage or stages. Unchecking this option will restrict the search to files not marked for prefetch.

5. Check or uncheck the desired **Find Options**:
   - **Match case** – With this checked, “Jukebox” will not match “JUKEBOX.”
   - **Match whole item only** – With this checked, “Photoshop” will not match “Adobe Photoshop.”

NOTE

The Search feature searches only the File and Registry views. The other views are special cases of these views. For example, Startup items also appear in either the File or Registry views. Search will locate those items within those two views.

6. Enter a search string in the **Find what** field.
7. Click the **Find** button.

The results of the search will be displayed in the **Find Results** window (see The Find Results Window below) located in the lower half of the Jukebox Studio window by default.

The Find Results Window

The results of a search are displayed in the **Find Results** window and grouped by asset type. The window is located in the lower half of the Jukebox Studio window by default but you can move it around by dragging its title bar to another edge of the main window or anywhere else to keep it floating.

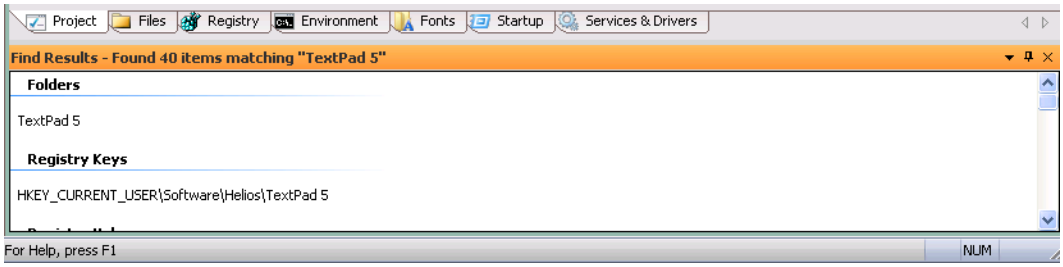


Figure 7-34: Find Results Window

From within the window, its is possible to jump to a specific asset in its associated view. Additionally, the **Find Results** window allows multiple assets to be selected and **Properties** can be changed for all selected assets at once as long as they are of the same asset type.

It is also possible to selectively remove results from the window if it is too crowded with results of no interest.

To jump to a specific asset:

- Double-click the asset you wish to jump to.

- OR -

- Right-click the asset you wish to jump to and select **Jump To**.

To change properties:

1. Select the assets you wish to change.

NOTE

Use Ctrl-click to select multiple assets. To select a range of consecutive assets, click the first asset in the range then drag the mouse cursor to the last asset. To select multiple ranges, press Ctrl while selecting each range.

2. Right-click the selection and select **Edit**.
3. Update the properties and apply the changes.

For example, to set security on all EXE files, find ***.exe** and select all of the files. Then select **Edit** and change the security options for the selected files.

To remove assets from the results list:

1. Click the assets you wish to change.

NOTE

Use Ctrl-click to select multiple assets. To select a range of consecutive assets, click the first asset in the range then drag the mouse cursor to the last asset. To select multiple ranges, press Ctrl while selecting each range.

2. Right-click the selection and select **Remove from list**.

To clear the results list:

- To remove all results from the list, right-click any asset and select **Clear list**.

About Search Types

Jukebox Studio has three types of searches:

- **Normal** – normal searches look for an exact match on a string of characters.
- **Use wildcards** (see Using Wildcards below).
- **Use regular expressions** (see Using Regular Expressions below).

Using Wildcards

These search queries uses the standard wildcard characters (* or ?) to help find patterns. An asterisk will search for any number of characters and the question mark will match 0 to 1 characters.

Using Regular Expressions

Regular expressions are a powerful tool for pattern matching.They are capable, for example, of finding a string of exactly six characters, beginning with an uppercase letter, followed by four lowercase letters and a single digit, and only at the beginning of a longer string. Regular expressions enable you to search with a high degree of precision and maximum flexibility.

Because regular expressions are an advanced topic beyond the scope of this guide, they are recommended for advanced users. The **Helper** menu next to the search box will get you started, but if you want to learn more, a good place to look would be the tutorial at regular-expressions.info.

Globally Changing Asset Dispositions

This tool enables you to change the disposition layer for all assets of a specific type.

CAUTION

This is an extremely powerful feature that will makes changes to every file, registry key, font, or driver. You should save your project before using this tool.

NOTE

The **Global Disposition Settings** dialog partially takes the place of the Virtual Settings option in previous version of Jukebox Studio. The other part of this functionality is handled by the **Sandboxing** screen in the **Settings** dialog. See Sandboxing Settings on page 63 for more information.

To Globally Change Dispositions:

1. Click the **Edit** menu and select **Global Disposition Settings**. The **Global Disposition Settings** dialog appears.

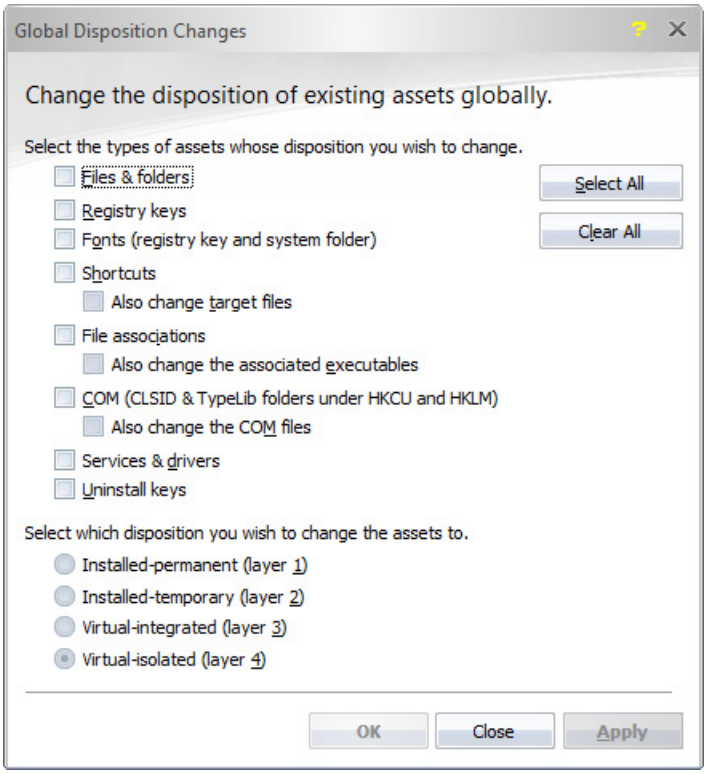


Figure 7-35: Global Disposition Settings

2. Check the types of asset that will be affected:
   - **Files** – This will affect all files under the Files view.
   - **Registry Keys** – This will affect all registry keys in the Registry view.
   - **Fonts** — This will affect all fonts in the Fonts view. Note that Fonts are also shown in the Files view, so changes to Files also affects fonts.

- **Short Cuts** — This will affect shortcuts on the Desktop or Start Menu.
- **File Associations** — This will affect any file type associations created by the cloudified application. You also have the option of checking a second box to change the disposition of the executable files associated with the file types.
- **COM** — This will affect registry keys for COM-based libraries. Check the second box to also change the disposition of the related image files.
- **Services and Drivers** — This will affect registry keys and image files for assets in the Services & Drivers view.
- **Uninstall Keys** — This will affect any entries for the cloudified application on the Windows Remove Program dialog.

3. Select the new disposition layer.
4. Click **OK** or **Apply**. The Apply button leaves the Global Disposition Settings open, so you can make more changes.

Understanding Color Coding in Views

If you see an asset displayed in color in one of the workspace views, that indicates something about the status of that asset. If you hover your mouse of the asset, you should see brief description of the conflict in a tool tip message.

The following table shows how assets are color-coded in Views.

Color/Attribute	Asset Type	Meaning
Black	All Assets	Normal
Grey	Template Folders	Empty
Grey Italic	Files Root Folders Sub-folders Registry Keys Registry Values	Excluded
Blue	Files Folders Registry Keys Registry Values	New (After capture)[1]
Magenta	Files Folders Registry Keys Registry Values	Changed (After capture)[2]
Red	Files Folders Registry Keys Registry Values	Skipped during cloudifying.
	All Assets	Missing asset or conflict.[3]
Bold	Folder	Working Folder (or ancestor).
	File	Command Line

1. This does not apply to manually added assets.
2. This does not apply to manually changed assets.
3. See the table below details about the probable cause of the conflict.

Conflict Color Coding

The following table details the various conditions that will result in a conflict (red) condition.

View(s)	Color-Coded Field	Meaning
Font StartUp Services & Drivers	File Command Line	File associated with asset is not in the project. **Note:** This may be okay if the file is known to always be on the end user system and at the same location (e.g. notepad.exe) **Note (exception):** If the file is in the project but excluded, it is shown in grey italic (as indicated in the previous table). This is the only case where a non-red color indicates a potential conflict.
Font	Action	Fonts are set to be registered, but the font will not be visible to the system because the Fonts key is set to layer 4.
	Action + Disposition	Fonts are set to be registered, but the font will not be visible to the system because the font file is set to layer 4.
Startup (Registry)	Disposition	Registry Run (or RunOnce) key asset cannot be started by the system because the key is set to layer 1 or 2, but its target command line is set to layer 3 or 4 and will not be visible to the system.
StartUp (Files)	Name + Disposition	One of these two conditions exists: • Shortcut file does not exist on the system. • Startup shortcut cannot be started by the system because it is set to layer 1 or 2, but its target command line is set to layer 3 or 4 and will not be visible to the system.
Services and Drivers	Name	Service is set to restart or reboot automatically upon failure. This will cause the service to always restart when the application is shut down from the Player, which will put it back into the Running state. Toggling this service's Failure Actions to OFF will fix this issue.
	Action	Action cannot be performed on a service whose type is not specified.
	Type	Service type is not specified.
	Interactive	Interactive setting is not specified.
	Start	Start type is not specified.
	Error Control	Error control type is not specified.
	Image Disposition	Driver is set to start on boot or system start, but its service key and/or image file is set to layer 3 or 4 and will not be visible to the system at that time.
	Both Key and Image Dispositions	Image file layer is greater than service key layer, which can yield inconsistent behavior. **Note:** One exception -- it is okay for the service key to be at layer 3 and the image file at layer 4.

Chapter 8

Modifying an Appset

This chapter covers these topics:

- **Overview**
- **Opening an Appset file**
- **Editing an AIB file**
- **Saving an Appset File**
- **Editing Token files**

Overview

Appsets can be modified by editing either the Application Installation Blueprint file (**.aib**) and Token files (**.tok**). The AIB file consists of metadata about the appset's file and registry settings, and the Token file consists of license key data and information provided by the software publisher.

The purpose of modifying the AIB or TOK file is to allow for changing security settings, disposition, prefetch settings, file attributes, as well as file names and locations. Modifying AIB files is particularly useful when modifying appset settings during Phase 4 of the cloudifying process (Testing and Optimizing the Appset). Modifying the AIB file saves the extra time that would be spent re-cloudifying the appset after each testing iteration.

NOTE

New files can not be added to an appset (i.e. AIB or TOK file). A new appset must be created or an appset must be patched if new files are required.

Additionally, the EULA or Icon file for that appset can also be modified by following the Opening an Appset file on page 118 and Saving an Appset File on page 144 instructions.

CAUTION

All modifications made by editing the TOK and/or AIB file are not saved in the original STW file. Thus when patching an appset, any of these modifications would be lost on new patch revisions. To avoid this inconsistency, it is recommended to make modifications for testing only and reapply final changes to the original STW.

Opening an Appset file

Before an appset can be modified, it must first be extracted from the appset or **.stp** file.

NOTE

You can open a Read-Only token file for viewing, but it cannot be edited. In order to edit a token file, you must have read and write permission for the token file.

To open an appset's Token or AIB file:

1. Unzip the **.stp** file by renaming it as a **.zip** file, right-click it, and select **Extract Here**. For example, rename "test.stp" to "test.zip" and then extract the contents.

   The contents of the **.stp** file (**.stc**, **.aib**, **.tok**, etc.) is extracted to the folder you selected.
2. Double-click the AIB or Token file.

   The contents of the AIB or Token file appears on the left pane of the Jukebox Studio UI.

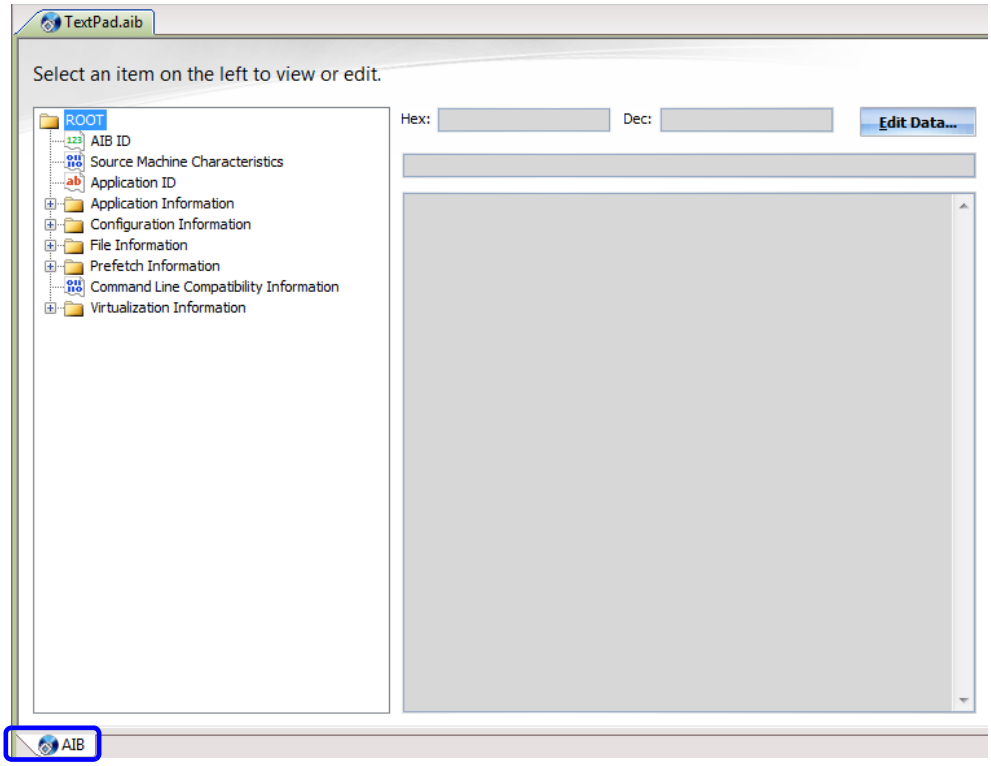


Figure 8-1: Example of an opened AIB file

You are now ready to modify the AIB or Token file.

Editing an AIB file

There are three types of Application Install Blueprint file modification: **modifying file settings**, **modifying registry settings**, **modifying minimum cache allocation**, and **modifying virtualization settings**.

Modifying File Settings

To modify file settings of an AIB file:

1. Open the AIB file. See Opening an Appset file on page 118 for instructions.

   The AIB view appears.

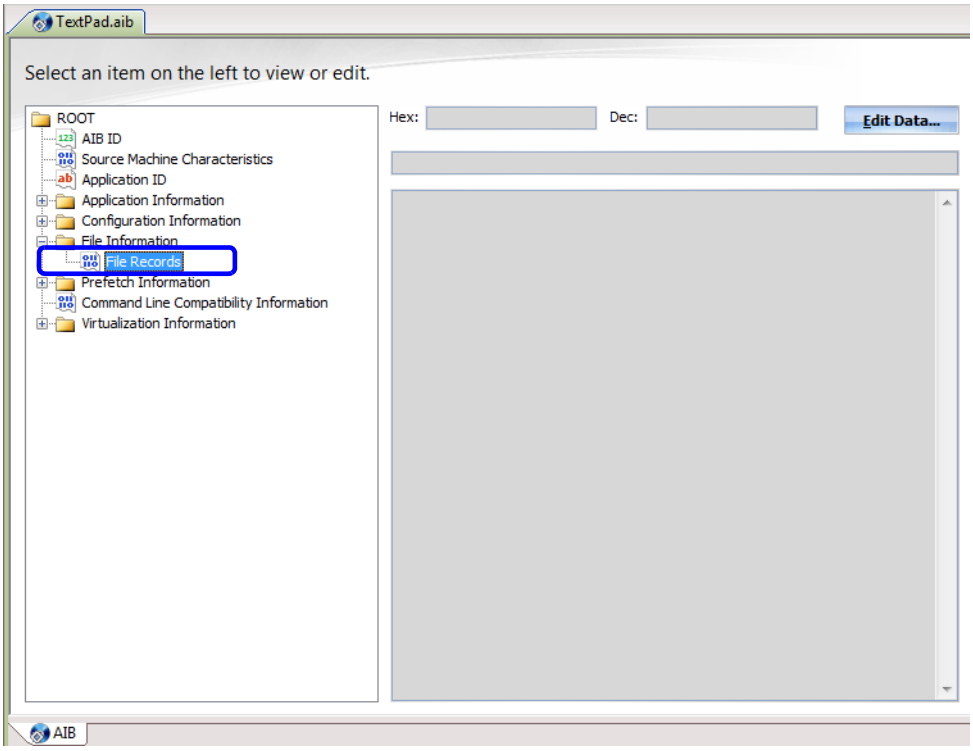


Figure 8-2: AIB File with File Information Group expanded

2. Expand **File Information**, and double-click **File Records**.

   The **Edit File Info Records** dialog appears.

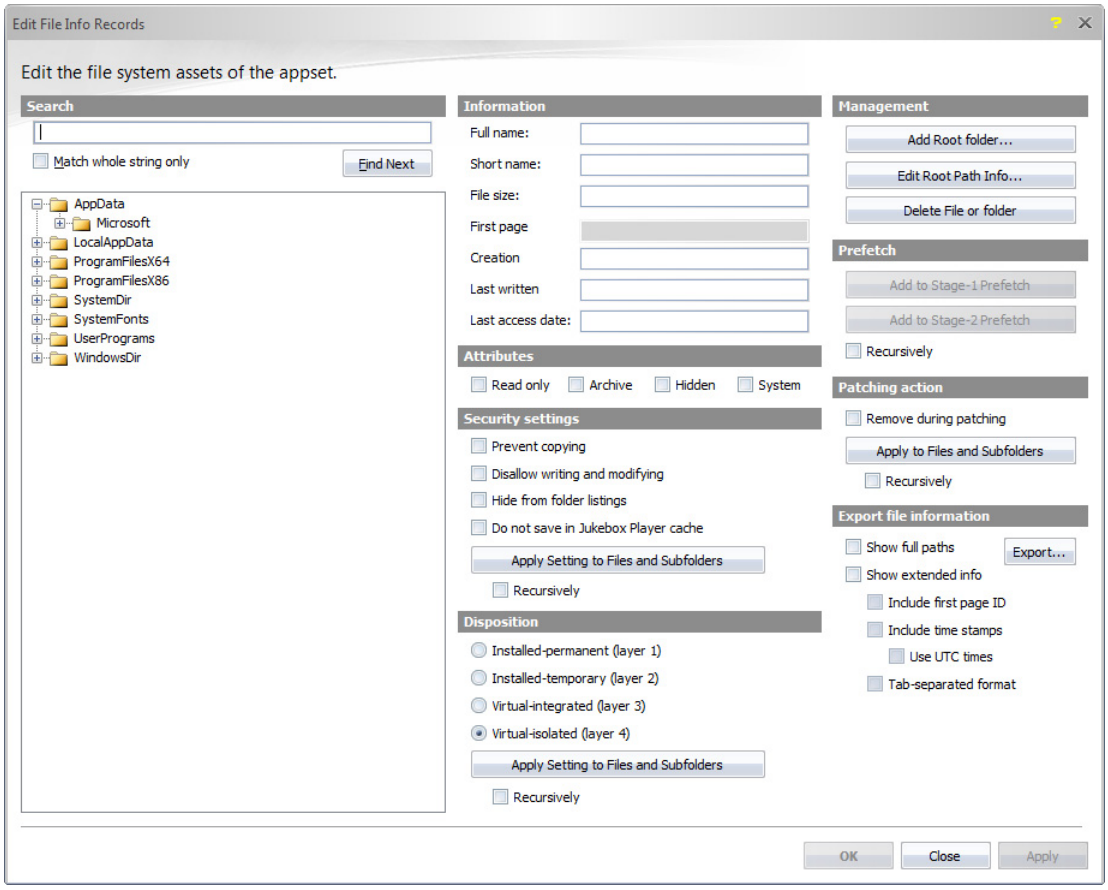


Figure 8-3: Edit File Info Records Dialog

3. Select a file or folder and make the appropriate setting changes. Most of these fields and options are explained earlier in this manual. Refer to references as cited.
   - Information:
     - **Full name** - The complete name of the selected assets.
     - **Short name** - The short name of the selected assets.

CAUTION

The **Short Name** must be identical to that of the **.stc** file associated with the Token File. If this field is incorrect, then the application will not run properly and the Jukebox Player will display an error message.

     - **File size**, **First Page Id**, **Creation date**, **Last written date**, **Last access date** - These fields are all auto-generated.
   - Attributes: **Read only**, **Archive**, **Hidden**, **System** - These options are the same as those found on the Windows file or folder Properties dialog.
   - Security Settings: **Prevent copying**, **Disallow writing and modifying**, **Hide from folder listings**, **Do not save in Jukebox Player cache** (see Editing Folder Properties on page 77).
   - Disposition: **Installed-permanent (layer 1)**, **Installed-temporary (layer 2)**, **Virtual-integrated (layer 3)**, **Virtual-isolated (layer 4)**. For more information see Editing Folder Properties on page 77.

- Management (buttons): **Add New File**, **Add New Subdirectory**, **Edit Root Path Info**, **Delete File or Directory** - These functions are as labeled.
- Prefetch:
  - **Add Directory to Stage-1 Prefetch** - Adds the current folder to the Stage-1 prefetch and adds Stage-1 prefetch to the appset, if necessary.
  - **Add Directory to Stage-2 Prefetch** - Adds the current folder to the Stage-2 prefetch and adds Stage-2 prefetch to the appset, if necessary.

4. Click **OK**.

Exporting File Information

Jukebox Studio can output a text file that lists every folder and file shown in this view, along with the option of showing attributes and timestamps.

Options

- **Show Full Paths** — If checked, the output file will contain the complete file path for each file. If not, the file will show a hierarchical tree view.
- **Show Extended Info** — If checked, the output file will contain attributes. If not, it will only contain the file/folder names and paths. If this is checked, the following options are available:
- **Include First Page ID** — If checked, the file will contain the page ID for each file.
- **Include time stamps** — If checked, the file will contain the creation, last modification, and last access timestamps for each file.
- **Use UTC times** — If checked, the timestamps will be written in Greenwich Meridian or UTC time, rather than local times. This option is not available if timestamps are disabled.
- **Tab-separated format** — If check, the file will be output in a tab-delimited format for importing into spreadsheet programs such as Microsoft Excel. If not, it will be formatted for printing or viewing.
- **Export** — Click this button to name and safe the file.

Searching for File Information

To locate a specific file:

1. Type a full or partial file or folder name into the Search box at the top of the **Edit File Info Records** dialog.
2. Click **Find Next**.
3. If a file or folder name matches the search, the first matching file or folder will be highlighted and its information displayed.
4. If this is not the correct asset, click **Find Next** until the desired file or folder is displayed.

Modifying Registry Settings

To modify the registry settings of an AIB file:

1. Open the AIB file. See Opening an Appset file on page 118 for instructions.

   The AIB view appears.

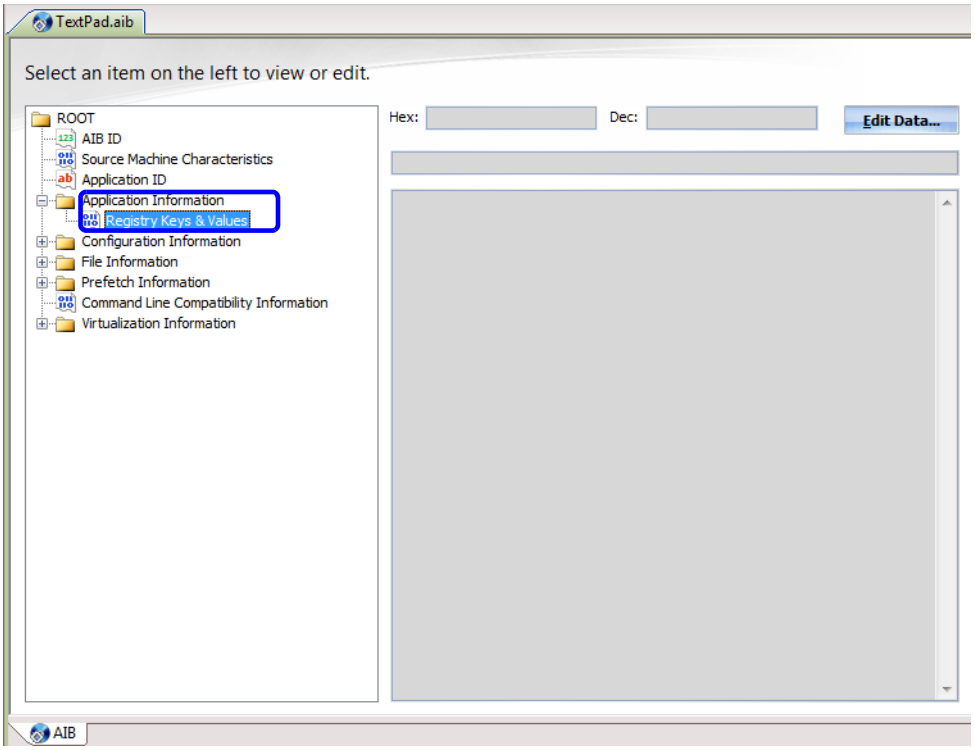


Figure 8-4: AIB File with Application Information Group expanded

2. Expand **Application Information**, and double-click **Registry Keys &Values**.
   The **Edit Registry Blob** dialog appears.

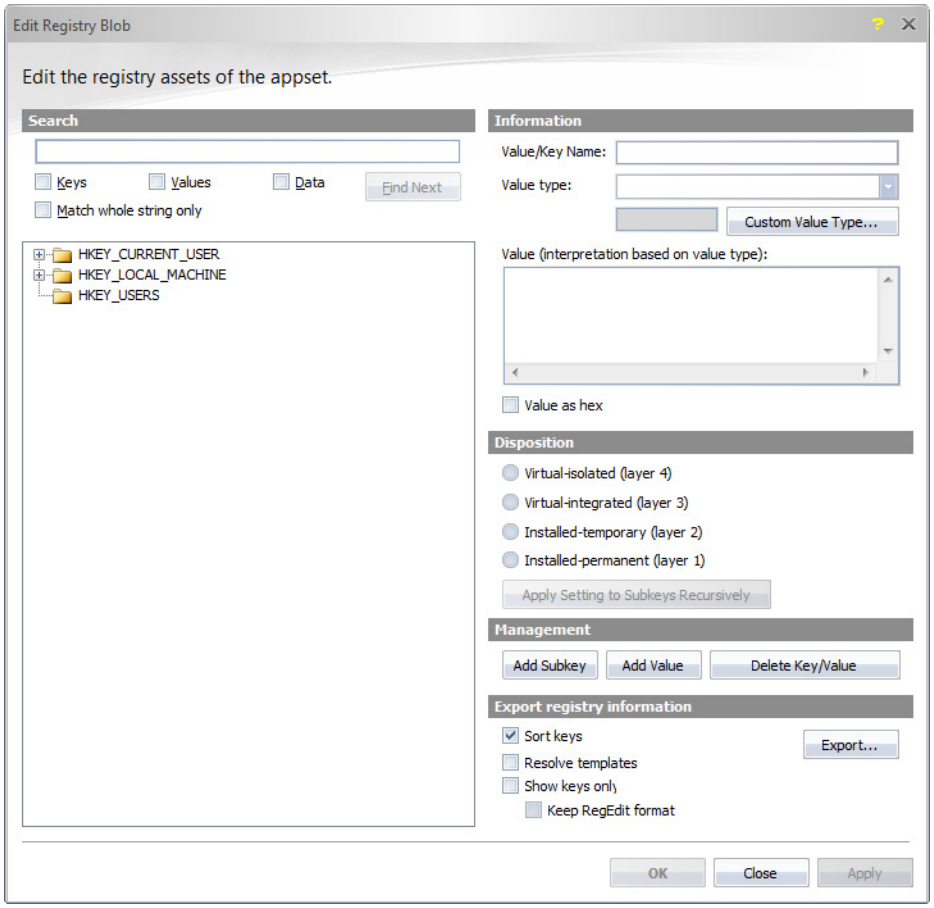


Figure 8-5: Edit Registry Blob Dialog

3. Select a key or value and make the appropriate setting changes. Some fields and options are explained earlier in this manual. Refer to references as cited.
   - Information:
     - **Value/Key Name** - The name of the registry value or key.
     - **Value type** - The drop-down menu provides a choice of value types.

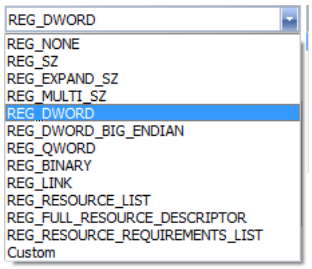


Figure 8-6: Value Type Drop-Down Menu

     - **Value** - Auto-generated.
     - **Value as hex** - Select this option to display the registry value as hexadecimal.

- Disposition: **Installed-permanent (layer 1)**, **Installed-temporary (layer 2)**, **Virtual-integrated (layer 3)**, **Virtual-isolated (layer 4)**. For more information, see Editing Folder Properties on page 77.
- Management (buttons): **Add Subkey**, **Add Value**, **Delete Key/Value**. These functions are as labeled.

Exporting Registry Information

Jukebox Studio can output a file, in RegEdit format, including all values, of every registry key in the project.

Options

- **Sort Keys** — If this is checked, the registry keys are written to the file sort alphabetically. If not, the keys are listed in the order found in the registry blob.
- **Show Keys Only** — If this is checked, the output file will be a flat listing of registry keys, one per line, without the values. If you check **Keep RegEdit Format**, each registry key will be in the form of a section header from a registry file, but without the values. In effect, Jukebox Studio will save a “blank” registry file.

Finding Registry Information

Jukebox Studio can locate individual keys and values using the search box at the top of the **Edit Registry Blob** dialog.

To locate a specific key or value:

1. Type the full or partial name into the Search box at the top of the **Edit File Info Records** dialog.
2. Check the boxes to indicate the type of registry asset that you want to search: **Keys**, **Values**, and **Data**.
3. Click **Find Next**.
4. If a registry asset matches the search, the first matching asset will be highlighted and its information displayed.
5. If this is not the correct asset, click **Find Next** until the desired asset is displayed.

Modifying Minimum Cache Allocation

To modify the minimum cache allocation settings of an AIB file:

1. Open the AIB file. See Opening an Appset file on page 118 for instructions.

   The AIB view appears.

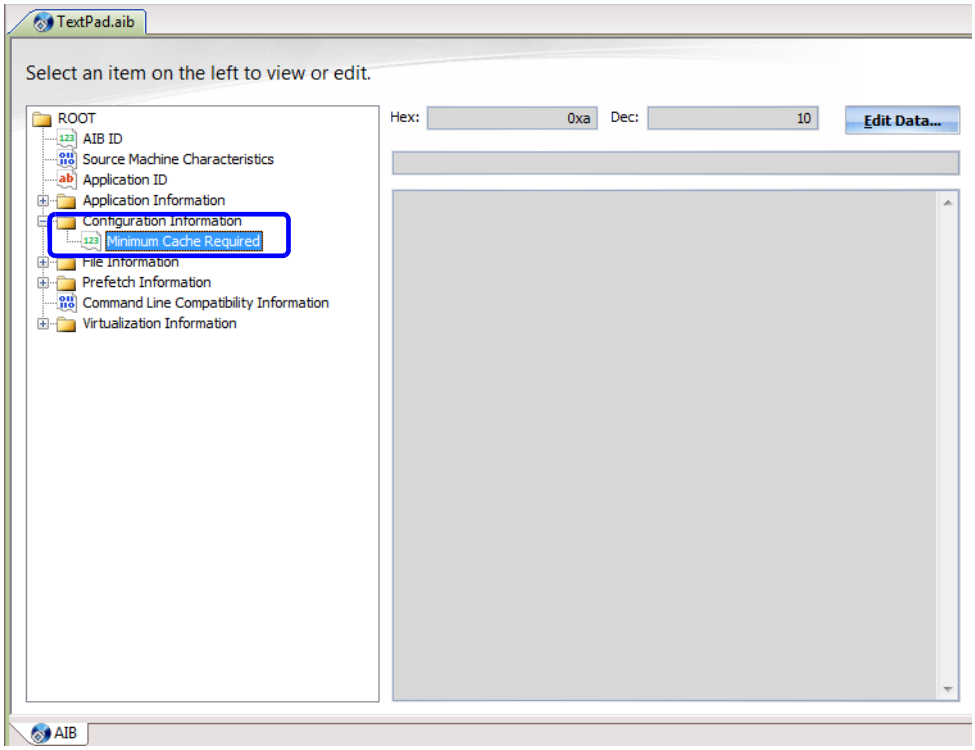


Figure 8-7: AIB File with Configuration Information expanded

2. Expand **Configuration Information**, and double-click **Minimum Cache Required**

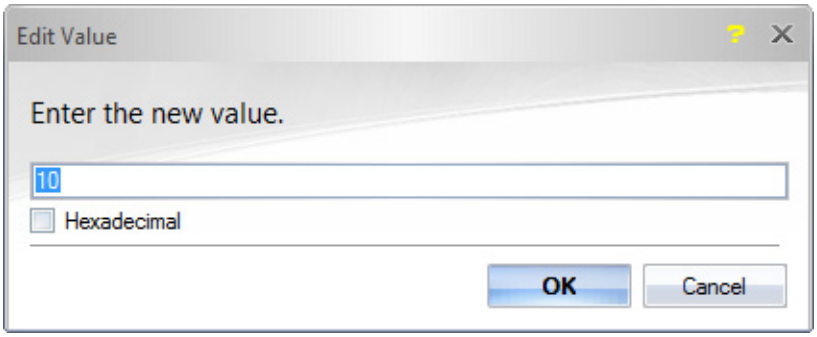


Figure 8-8: Enter the New Number Dialog

3. Enter the new value of 5 to 100, and click **OK**.

Editing Virtualization Information

To modify the virtualization settings of an AIB file:

1. Open the AIB file. See Opening an Appset file on page 118 for instructions.

   The AIB view appears.

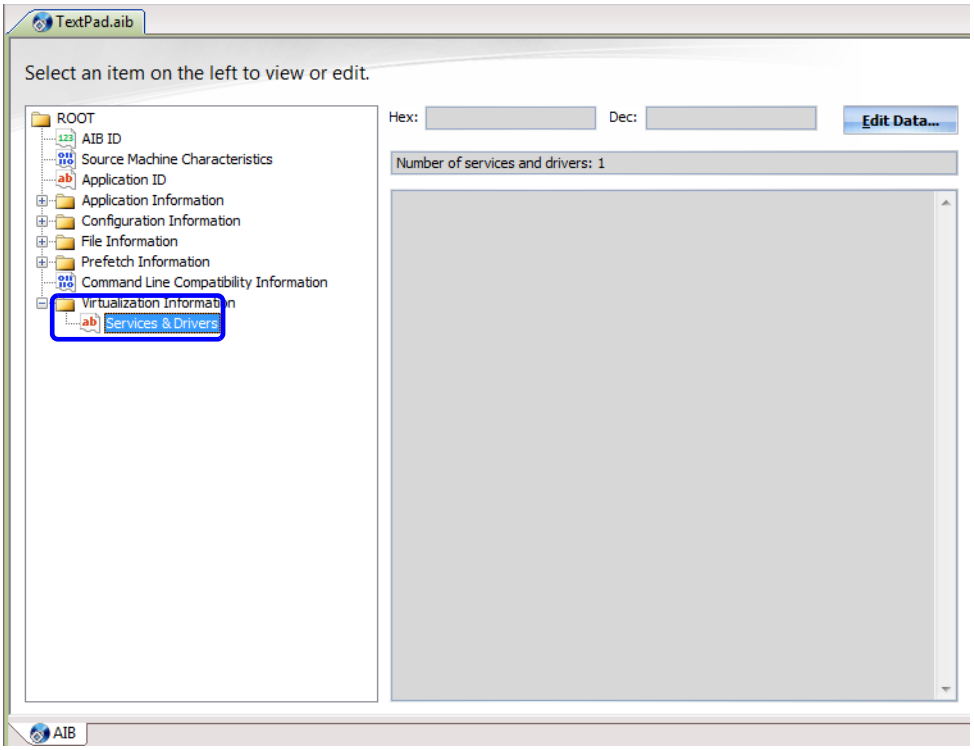
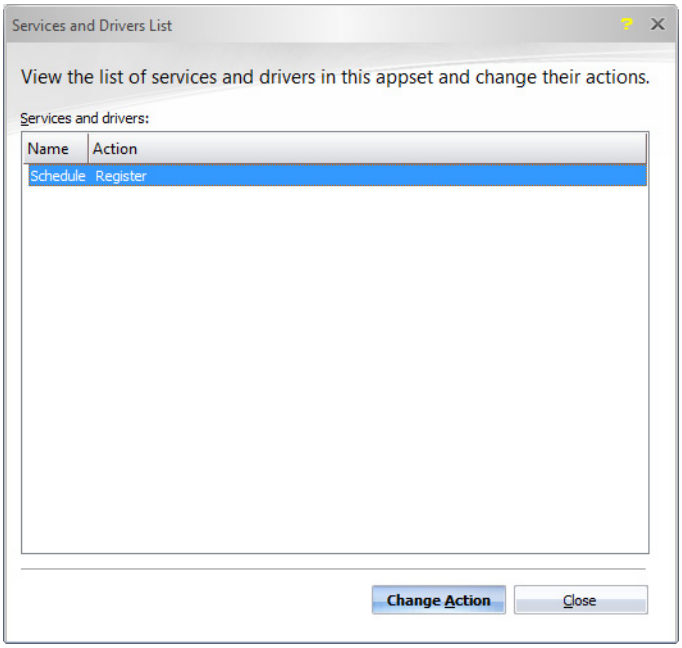


Figure 8-9: AIB File with Virtualization Information Group expanded

2. Expand **Virtualization Information**, and double-click **Services & Drivers**. Jukebox Studio displays the Services and Drivers List.



3. Select the desired service or driver and click **Change Action**. Jukebox Player displays the Change Service or Driver Virtualization Action dialog.

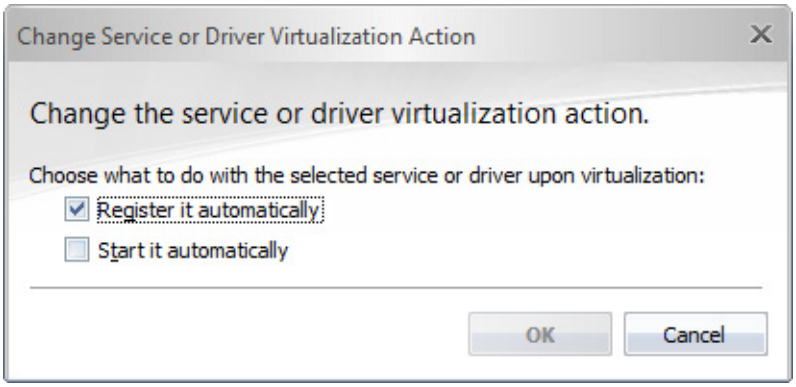

Change Service or Driver Virtualization Action
Change the service or driver virtualization action.
Choose what to do with the selected service or driver upon virtualization:
Register it automatically
Start it automatically
OK
Cancel

Figure 8-10: Change Service or Driver Virtualization Action Dialog

4. Select how you want Jukebox Player to handle the selected service and driver by checking or unchecking the boxes. For more information on these settings, see Virtualization Settings on page 62.

Adding Prefetch Files

From the AIB editor, it's possible to add prefetch files that were captured by Jukebox Player. See To capture and add a prefetch: on page 50 for the steps to follow. The procedure is identical.

Viewing Prefetch Information

To view the prefetch statistics:

The AIB file can show you the size of your Prefetch pages.

1. Open the AIB file. See Opening an Appset file on page 118 for instructions.

   The AIB view appears.

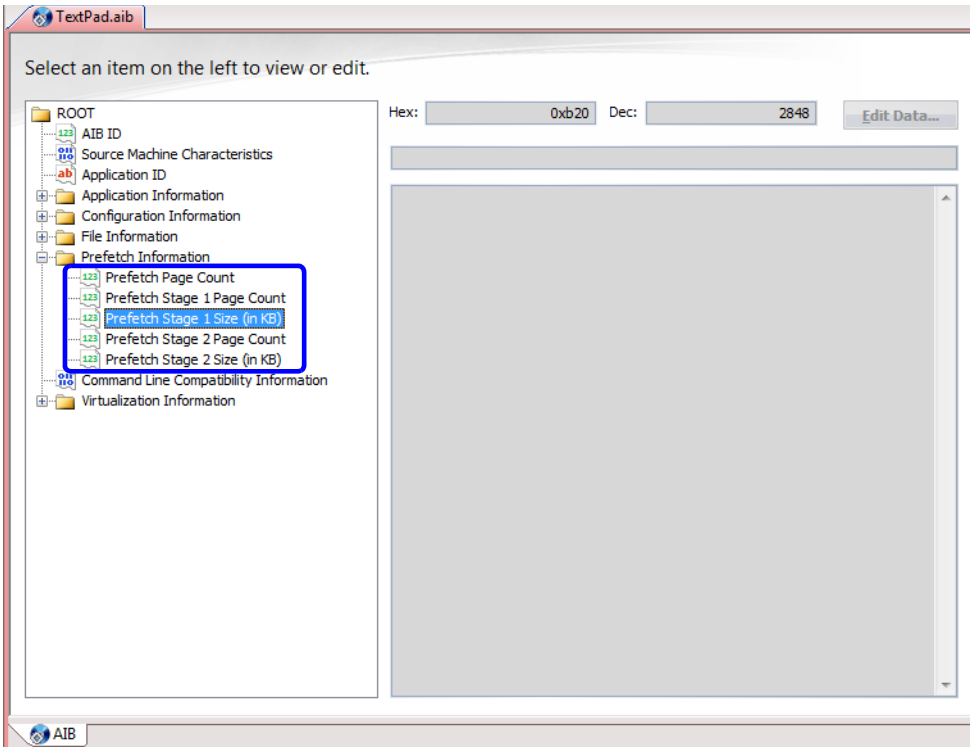


Figure 8-11: AIB File with Prefetch Information Group expanded

2. Select any of the following the see the number of pages of each type, along with their . Each page can be up to 32 kilobytes, but it could be smaller due to compression.
   - **Prefetch Page Count** – The total number of prefetch pages (both stage-1 and stage-2).
   - **Prefetch Stage 1 Page Count** – The number of stage-1 prefetch pages.
   - **Prefetch Stage 1 Size (in KB)** – The size in kilobytes of the stage-1 prefetch pages.
   - **Prefetch Stage 2 Page Count** – The number of prefetch pages loaded after the application has launched (stage-2).
   - **Prefetch Stage 2 Size (in KB)**– The size in kilobytes of the prefetch pages loaded after the application has launched (stage-2).

   If there are files assigned to more than two prefetch stages in the cloudified, those additional prefetch stages will be shown here, too.

Editing Compatibility Settings

To modify the Compatibility Settings in an AIB file:

1. Open the AIB file. See Opening an Appset file on page 118 for instructions.

   The AIB view appears.

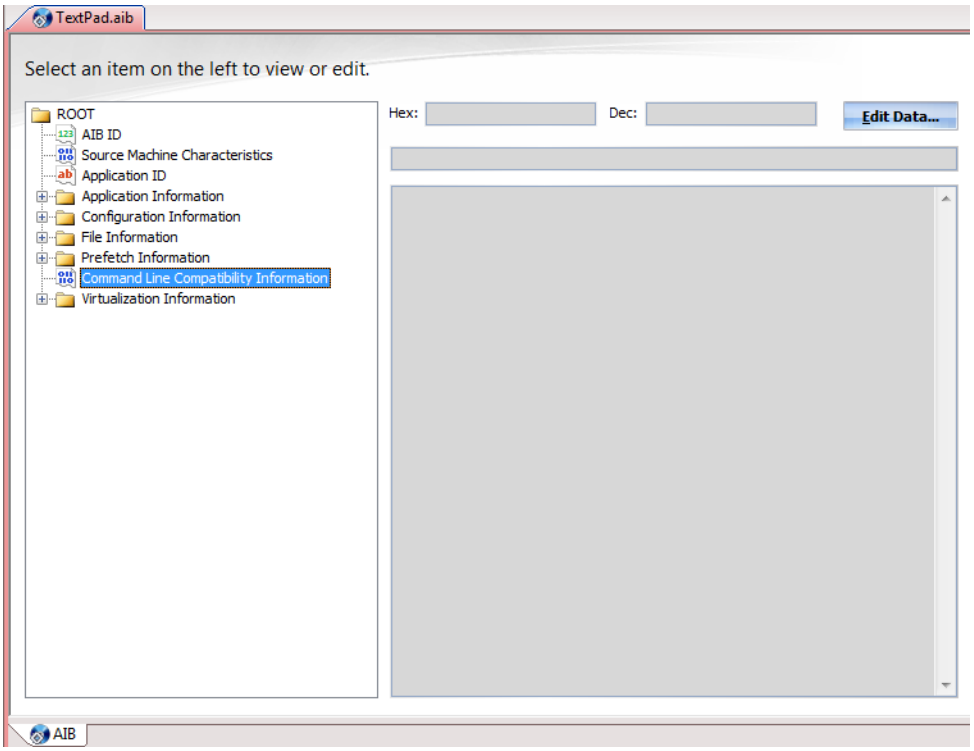
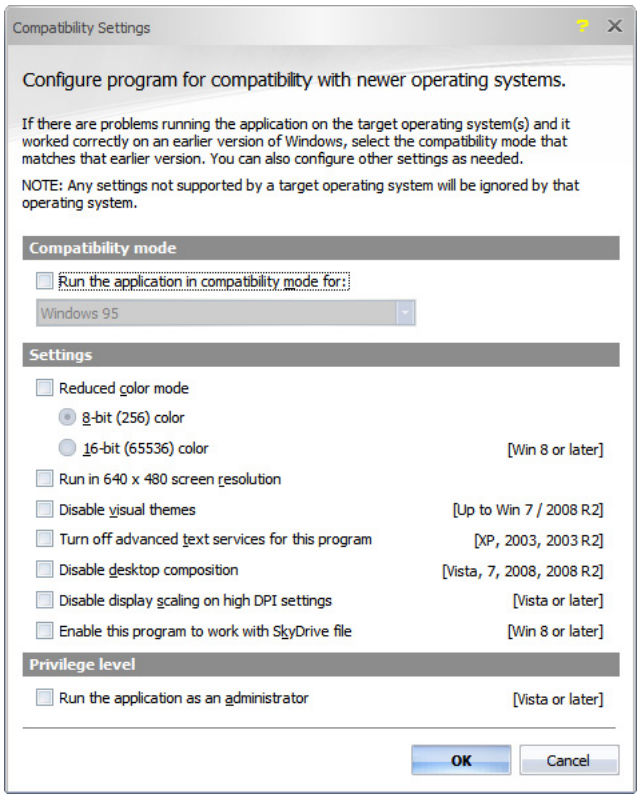


Figure 8-12: AIB File with Command Line Compatibility Information selected

2. Double-click **Command Line Compatibility Information**. Jukebox Studio displays the Compatibility Settings dialog.


Compatibility Settings
Configure program for compatibility with newer operating systems.
If there are problems running the application on the target operating system(s) and it worked correctly on an earlier version of Windows, select the compatibility mode that matches that earlier version. You can also configure other settings as needed.
NOTE: Any settings not supported by a target operating system will be ignored by that operating system.
Compatibility mode
Run the application in compatibility mode for:
Windows 95
Settings
Reduced color mode
8-bit (256) color
16-bit (65536) color
[Win 8 or later]
Run in 640 x 480 screen resolution
Disable visual themes
[Up to Win 7 / 2008 R2]
Turn off advanced text services for this program
[XP, 2003, 2003 R2]
Disable desktop composition
[Vista, 7, 2008, 2008 R2]
Disable display scaling on high DPI settings
[Vista or later]
Enable this program to work with SkyDrive file
[Win 8 or later]
Privilege level
Run the application as an administrator
[Vista or later]
OK
Cancel

3. Edit the desired settings. For more information, see Compatibility Settings options on page 40.

Editing Token files

There are seven types of Token file modification: **modifying an application value**, **modifying an OS compatibility setting**, **modifying a paging source**, **modifying a configurable appevent**, **modifying advanced settings flags**, **modifying an environment setting**, and **modifying security overrides**.

Modifying an Application Value

The Token values for the following can be changed:

- Full name
- Short name
- Description
- Minimum Client Version Required
- Game rating (ESRB Rating)
- System Requirements
- Reboot Flag
- Launch command line
- Working Directory

- Publisher Name

To change the value of a Token file:

1. Open the **.tok** file. See Opening an Appset file on page 118 for instructions. The Token view appears.

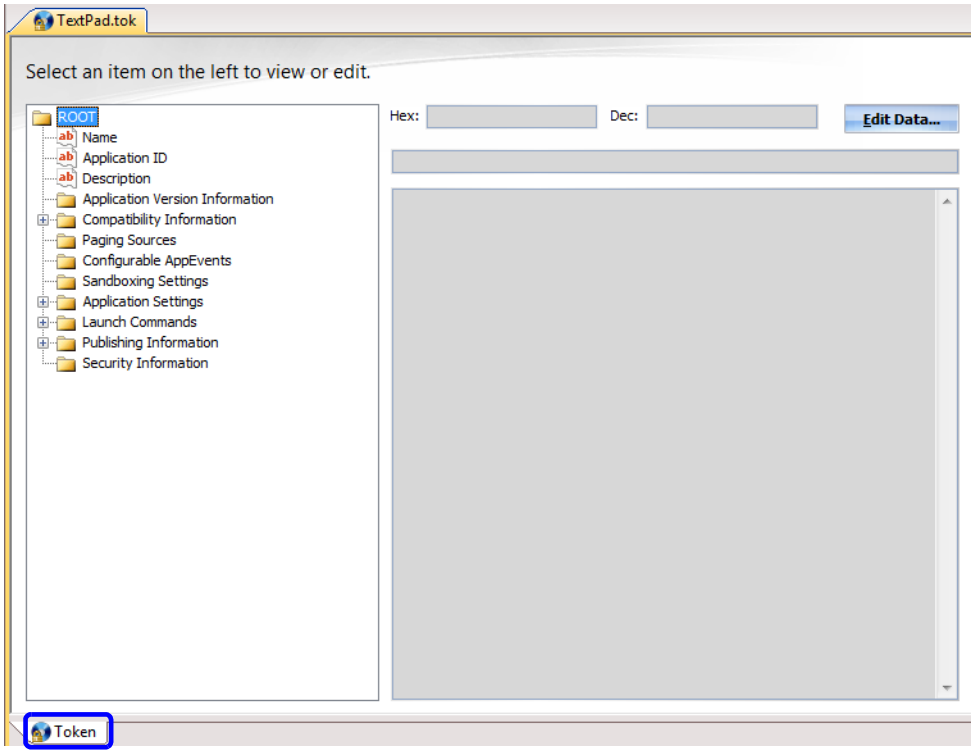


Figure 8-13: Token view

2. Double-click the Token value you wish to change (or select it, and click **Edit Value**).
   The **Enter the new value** dialog appears.

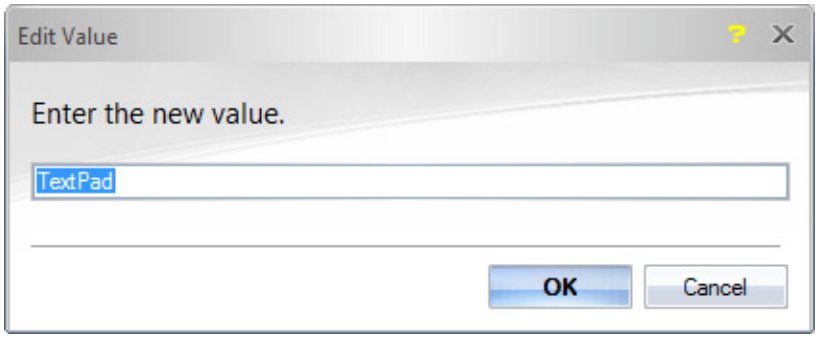


Figure 8-14: Enter the New Value Dialog

3. Enter the new value, and click **OK**.

Modifying Game Rating Settings

To modify the Game Rating of a Token file:

1. Open the **.tok** file. See Opening an Appset file on page 118 for instructions. The Token view appears.

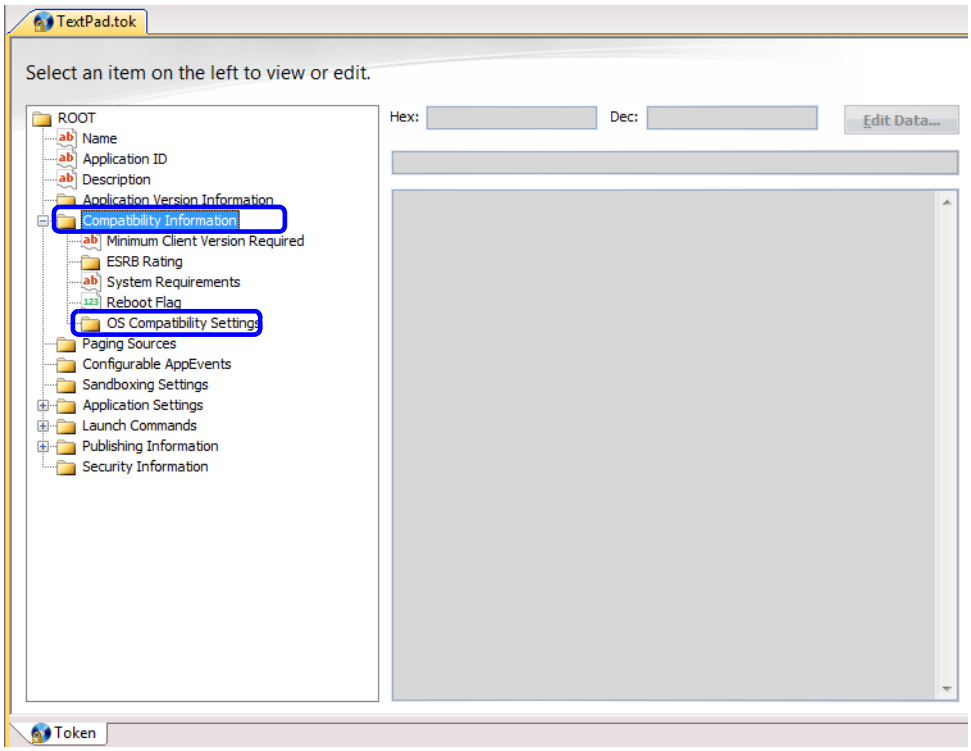


Figure 8-15: TOK File with Compatibility Group expanded

2. Expand **Compatibility Information**, and double-click **ESRB Rating**.

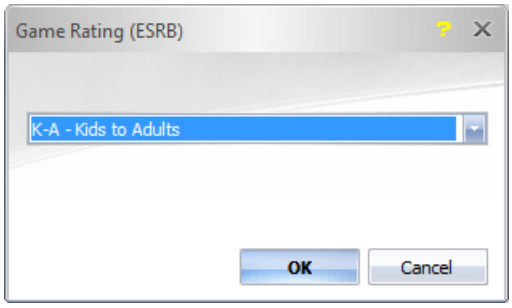


Figure 8-16: Game Rating Dialog

The **Game Rating** dialog appears.

3. Select a new game rating, and click **OK**.

Modifying OS Compatibility Settings

To modify the OS compatibility settings of a Token file:

1. Open the **.tok** file. See Opening an Appset file on page 118 for instructions. The Token view appears.

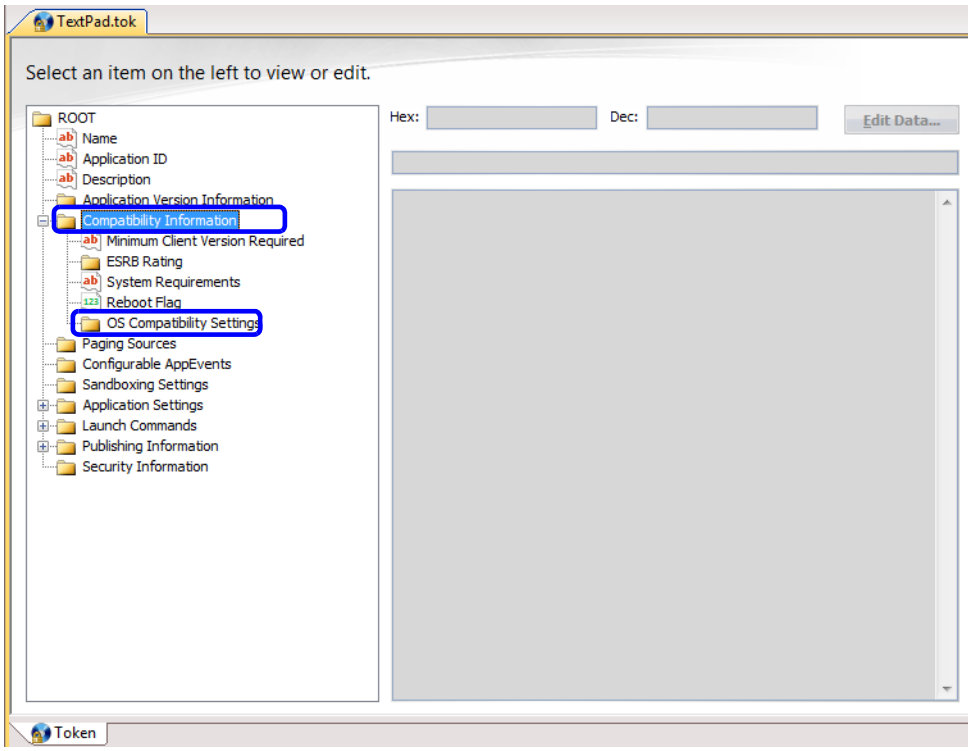


Figure 8-17: TOK File with Compatibility Group expanded

2. Expand **Compatibility Information**, and double-click **OS Compatibility Settings**.

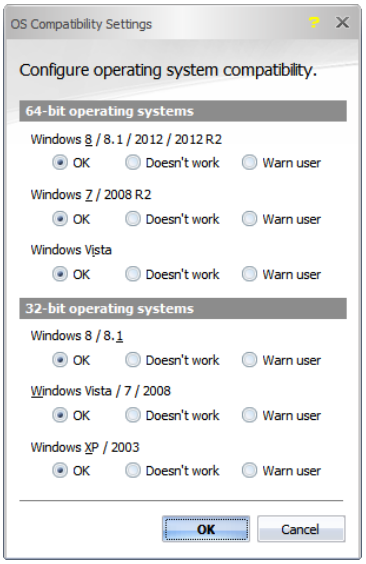


Figure 8-18: OS Compatibility Settings Dialog

The **OS Compatibility Settings** dialog appears.

3. Select the desired setting for each set of operating systems and click **OK**.

Modifying a Paging Source

To modify a Paging source settings of a Token file:

1. Open the **.tok** file. See Opening an Appset file on page 118 for instructions. The Token view appears.

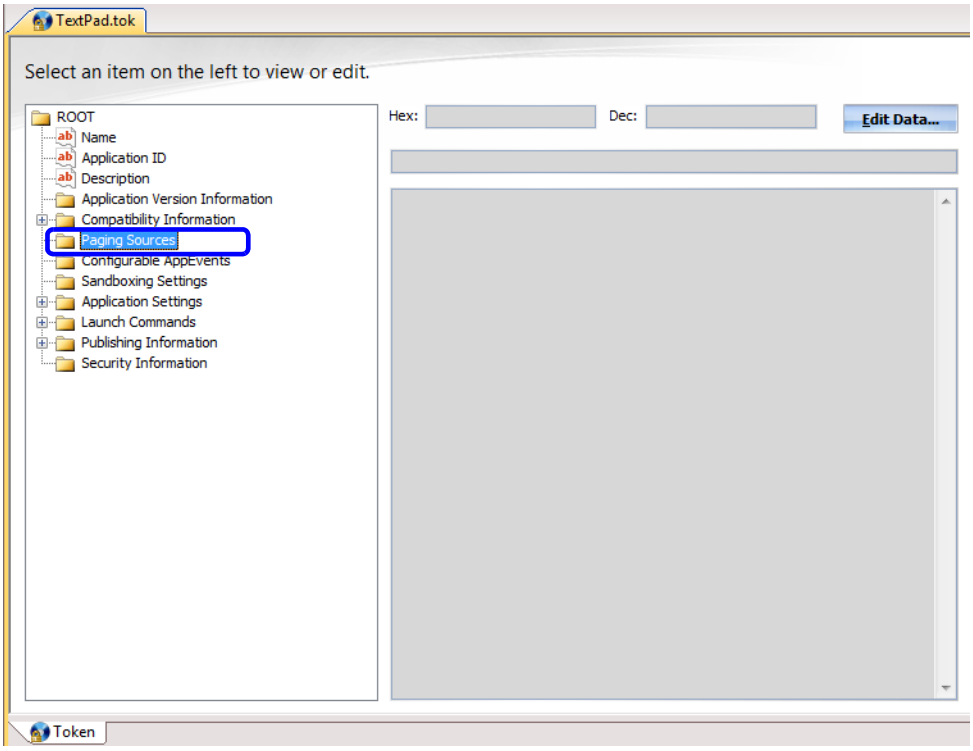


Figure 8-19: TOK File with Paging Sources shown

2. Double-click **Paging Sources**.
   The **Paging** dialog appears.

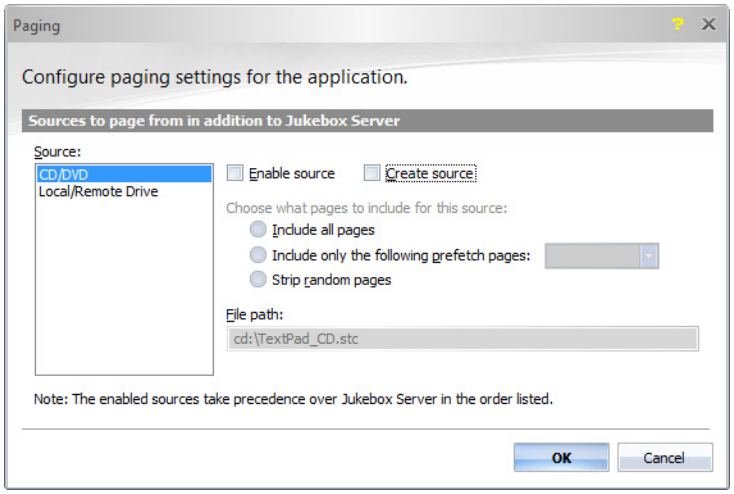


Figure 8-20: Paging Dialog

3. Disable or enable sources as necessary and click **OK**.

To create a new paging source:

1. Check Create source.
2. Then select one of three options:
   - **Include only the following prefetch page** – Indicates that the selected prefetch blocks will be made available at the source location and add them to a new source STC file.
   - **Include all blocks** – Indicates that all data blocks will be made available at the source location and add them to a new source STC file.
   - **Strip random blocks** – Indicates that except for a random few, all data blocks will be made available at the source location and add them to a new source STC file. These random blocks are marked with the security setting “Do not cache on local machine.” This is done for added protection as they must always be paged from another source.
3. Enter a file name and path for the new STC file.

Modifying a Configurable AppEvent

To add a Configurable AppEvent to a Token file:

1. Open the **.tok** file. See Opening an Appset file on page 118 for instructions. The Token view appears.

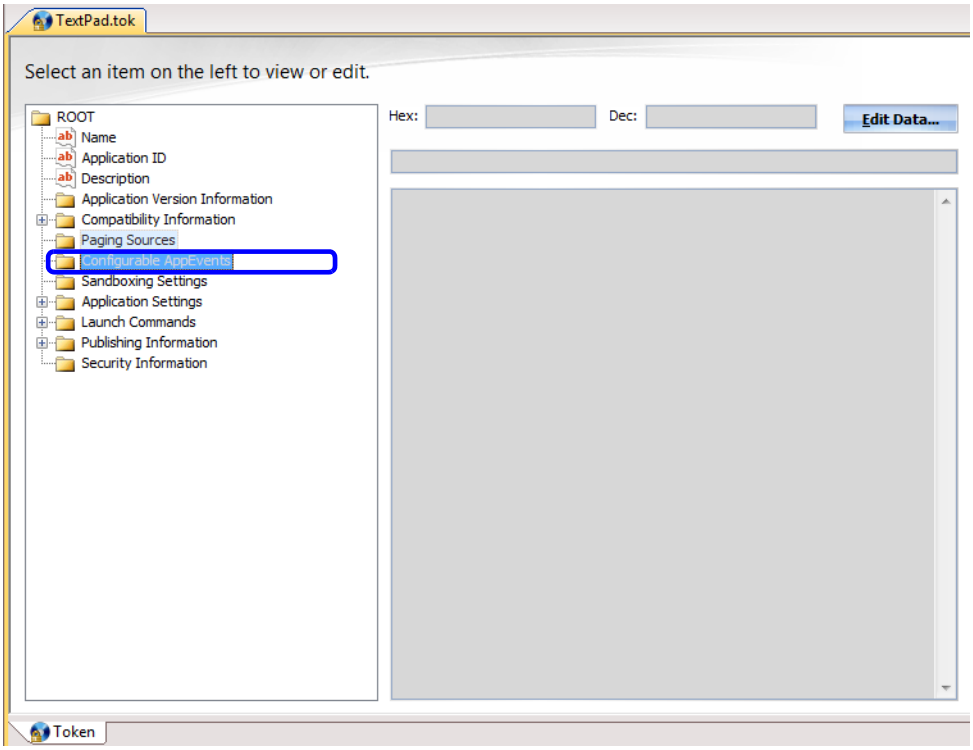


Figure 8-21: TOK File with Configurable AppEvents shown

2. Double-click **Configurable AppEvents**.

The **Configurable AppEvents** dialog appears.

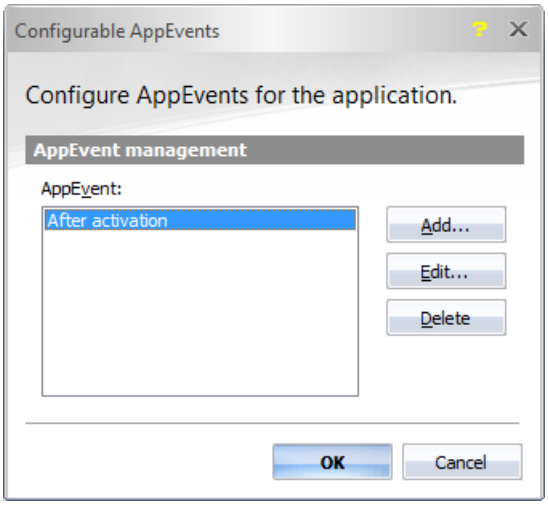


Figure 8-22: Configurable AppEvents Dialog

3. Click **Add**...
4. The Configurable AppEvents Settings dialog appears.

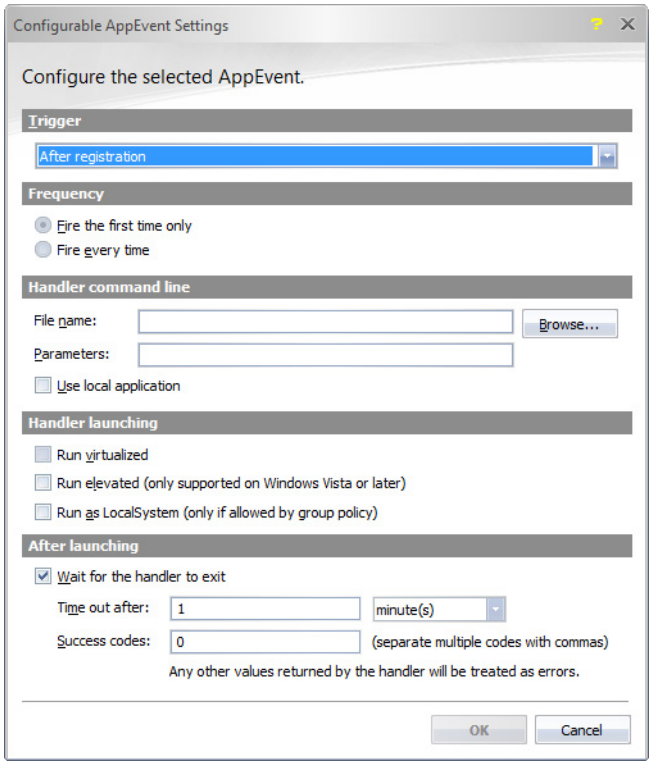


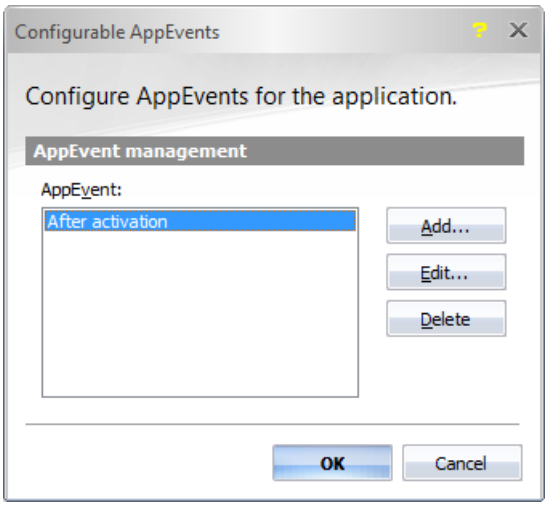
Figure 8-23: Configurable AppEvents Settings Dialog

5. Enter the settings for the new Configurable AppEvent. For more information, see To add an AppEvent on page 58.

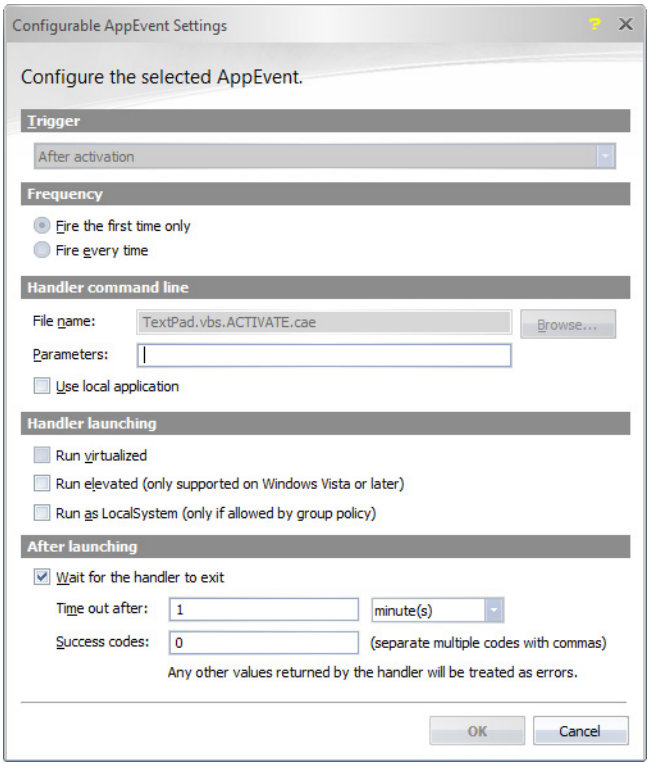
To Edit the Settings for an existing AppEvent

1. Open the **.tok** file. See Opening an Appset file on page 118 for instructions. The Token view appears.

2. Double-click **Configurable AppEvents**.
3. The **Configurable AppEvents** dialog appears.



4. Select an existing AppEvent and click **Edit...**
5. The Configurable AppEvents Settings dialog appears.



6. Edit the settings. For more information, see To add an AppEvent on page 58.

Modifying Sandboxing Settings

To modify the Sandboxing settings of a Token file:

1. Open the **.tok** file. See Opening an Appset file on page 118 for instructions. The Token view appears.

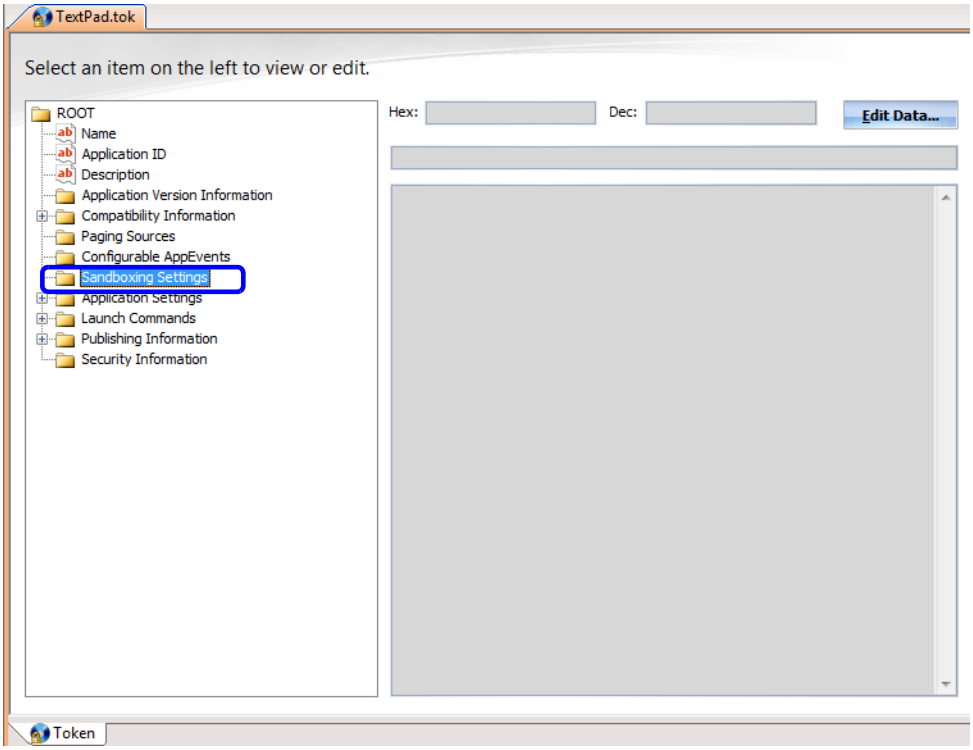
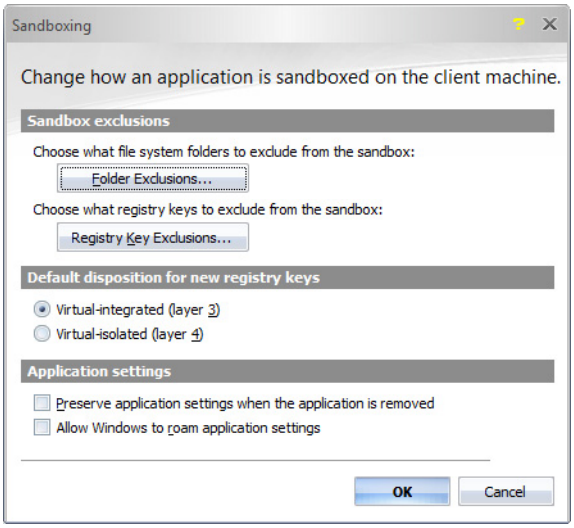


Figure 8-24: TOK File with Sandboxing Settings Selected

2. Double-click **Sandboxing Settings**.
3. The **Sandboxing Settings** dialog appears.


Sandboxing
Change how an application is sandboxed on the client machine.
Sandbox exclusions
Choose what file system folders to exclude from the sandbox:
Folder Exclusions...
Choose what registry keys to exclude from the sandbox:
Registry Key Exclusions...
Default disposition for new registry keys
Virtual-integrated (layer 3)
Virtual-isolated (layer 4)
Application settings
Preserve application settings when the application is removed
Allow Windows to roam application settings
OK
Cancel

4. For more information on using these settings, see Sandboxing Settings on page 63.

Modifying Advanced Settings Flags

To modify the extra settings of a Token file:

1. Open the **.tok** file. See Opening an Appset file on page 118 for instructions. The Token view appears.

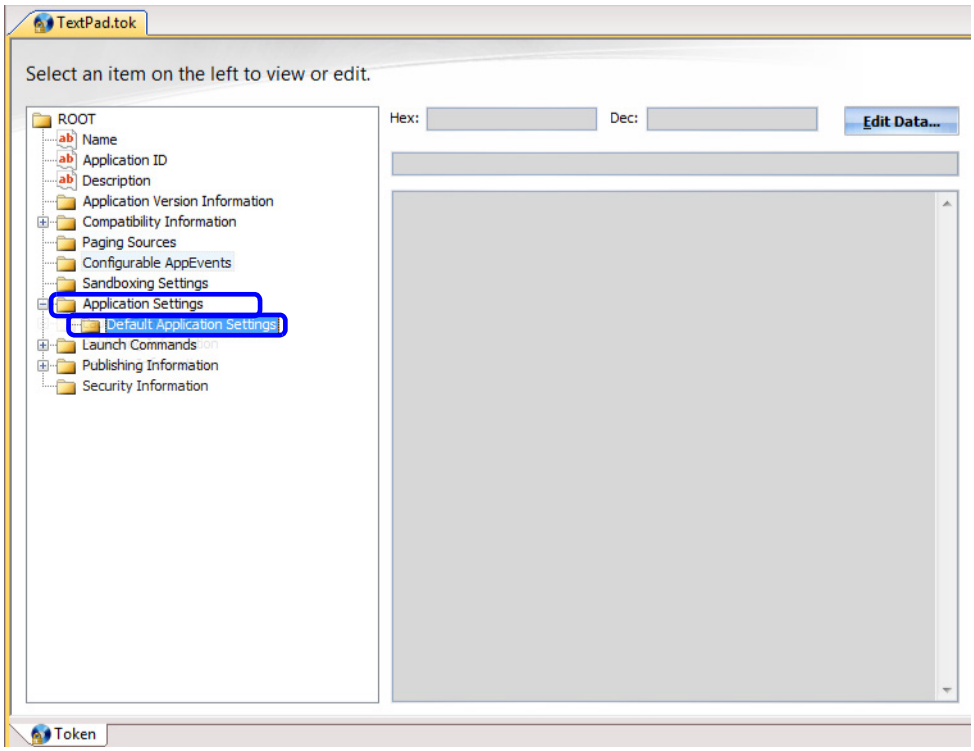


Figure 8-25: TOK File with Application Settings Group expanded

2. Expand **Application Settings**, and double-click **Default Application Settings**.
   The **Application Settings** dialog appears.

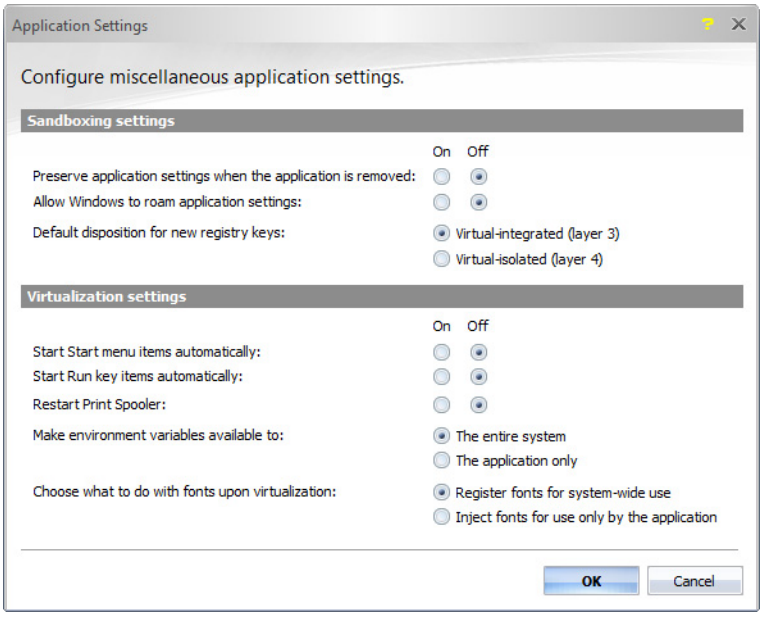

Application Settings
Configure miscellaneous application settings.
Sandboxing settings
On Off
Preserve application settings when the application is removed:
Allow Windows to roam application settings:
Default disposition for new registry keys:
Virtual-integrated (layer 3)
Virtual-isolated (layer 4)
Virtualization settings
On Off
Start Start menu items automatically:
Start Run key items automatically:
Restart Print Spooler:
Make environment variables available to:
The entire system
The application only
Choose what to do with fonts upon virtualization:
Register fonts for system-wide use
Inject fonts for use only by the application
OK
Cancel

Figure 8-26: Application Settings Dialog

3. Change the desired settings. For more information on the setting on this dialog, see Virtualization Settings on page 62 and Sandboxing Settings on page 63.

Modifying an Environment Setting

To modify an Environment setting of a Token file:

1. Open the **.tok** file. See Opening an Appset file on page 118 for instructions. The Token view appears.

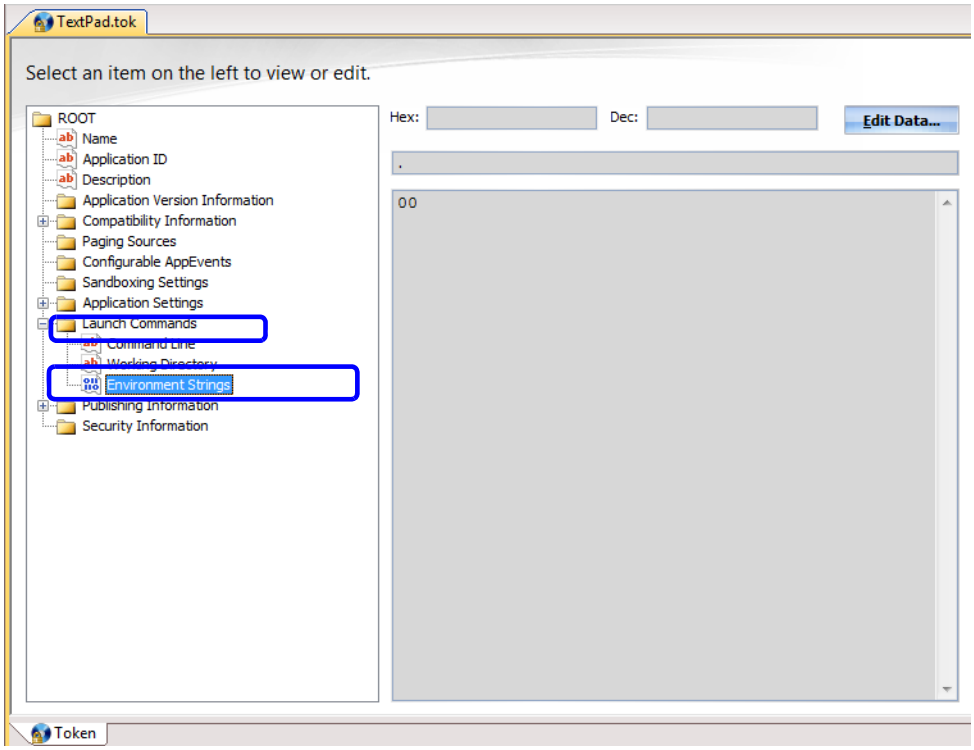


Figure 8-27: TOK File with Launch Command Group expanded

2. Expand **Launch Commands**, and double-click **Environment Strings**.

The **Edit Environment Strings** dialog appears.

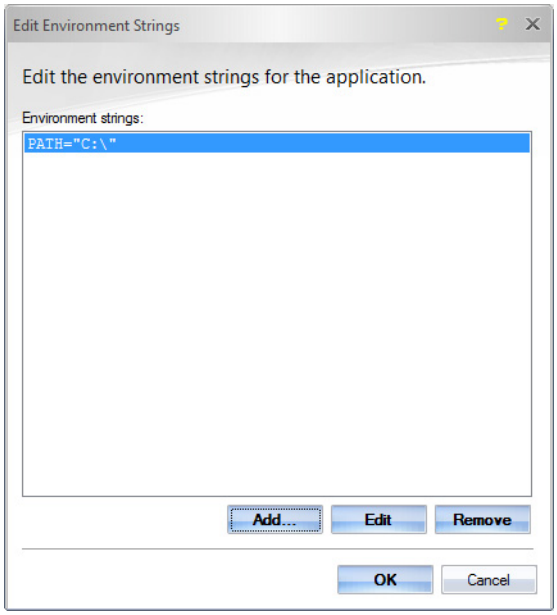


Figure 8-28: Edit Environment Strings Dialog

3. To add an Environment String, click **Add**.
   To edit an Environment String, click **Edit**.
   The **Enter the new value** dialog appears.

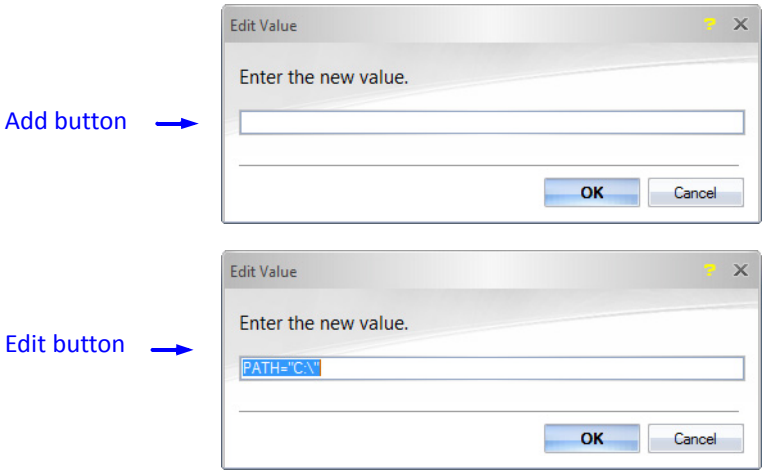


Figure 8-29: Enter the New Value Dialog

4. Enter the new value, and click **OK**.

To delete an Environment string:

1. Select the string from the **Edit Environment Strings** dialog.
2. Click **Remove**.

Modifying Security Override Settings

To modify the security override settings of a Token file:

1. Open the **.tok** file. See Opening an Appset file on page 118 for instructions. The Token view appears.

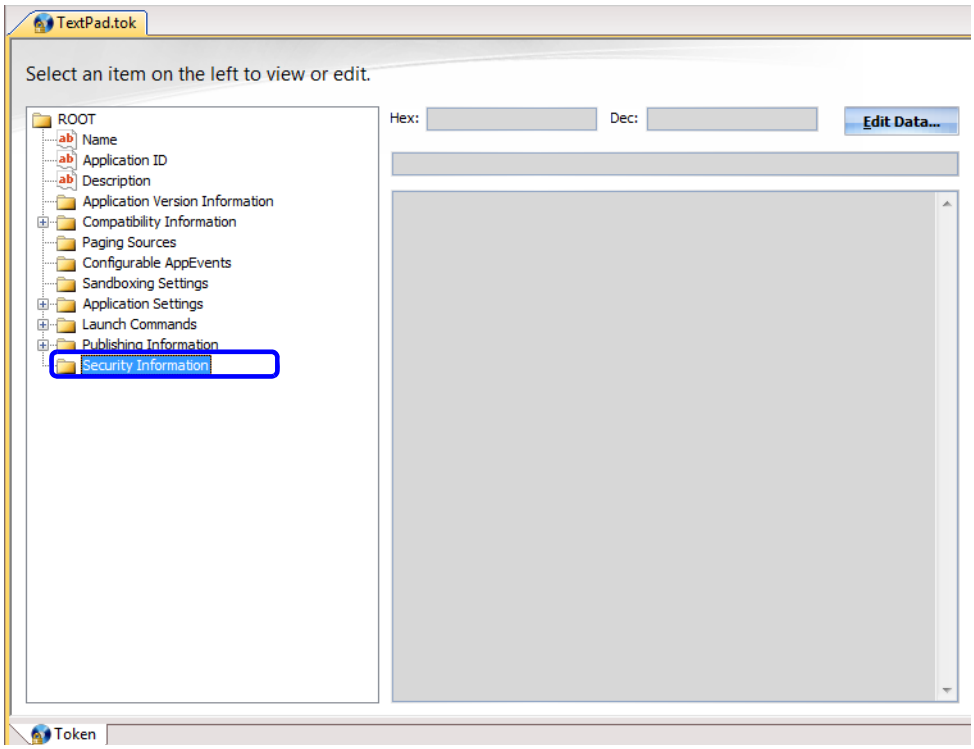


Figure 8-30: TOK File expanded

2. Double-click **Security Information**.

The **Security Overrides** dialog appears.

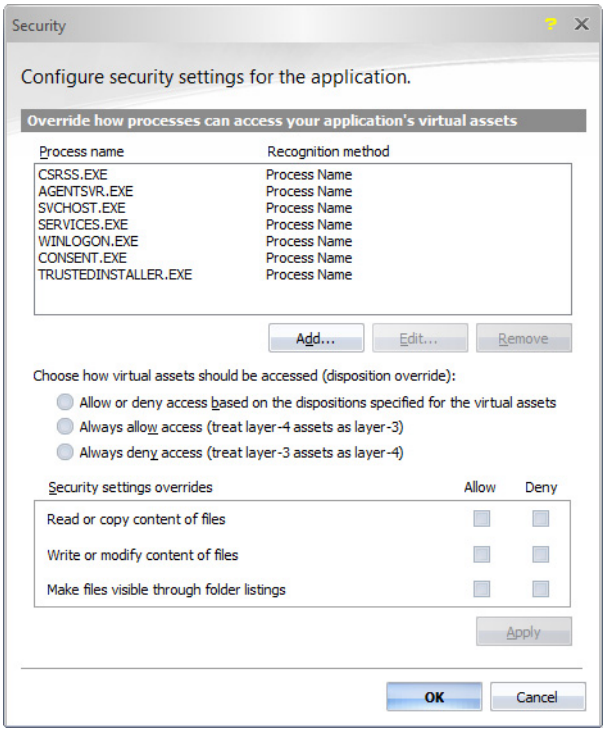


Figure 8-31: Security Overrides Dialog

3. Choose the desired security overrides and click **OK**.

Saving an Appset File

After the AIB and/or TOK file has been updated, they need to be saved back into the appset (**.stp**) file format. You could also use this procedure to update the EULA HTML file or icon file.

To accomplish this requires a third-party compression utility called 7-zip, which can be found here: http://www.7-zip.org/download.html. Please be sure to have this utility installed.

CAUTION

We recommend 7-Zip for its support of UTF-8 names and large files. It is a free open source utility. The built-in Windows file compression does not have this support and creating an appset this way could cause errors.

To save an appset file using the 7-Zip Command Line

1. Open a command prompt window.
2. Navigate to the location of the STC, AIB, TOK, ICO, or EULA files.
3. Enter the following command line:

   ```
   7z a -mcu=on MyAppset.zip MyAppset.tok MyAppset.aib MyAppset.stc MyAppset.ico
   ```

   Your appset may also include Configurable AppEvent files (.cae) files. These also needed to be included in the command line.
4. After 7-Zip creates the new ZIP archive, change the extension of the archive file to “stp.”

To save the appset using the 7-Zip GUI

1. Start 7-Zip.
2. In the 7-Zip explorer window, navigate to the folder containing your appset files.
3. Select the appset files.

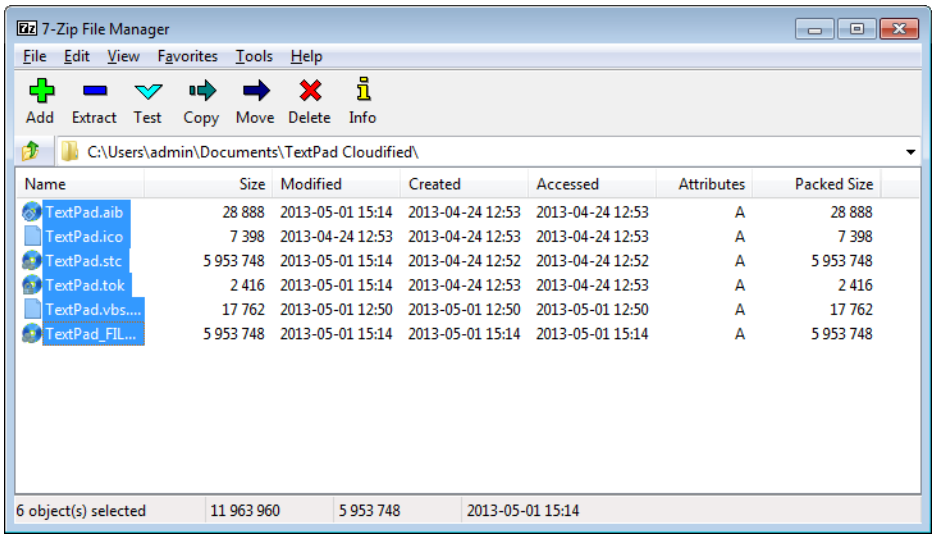
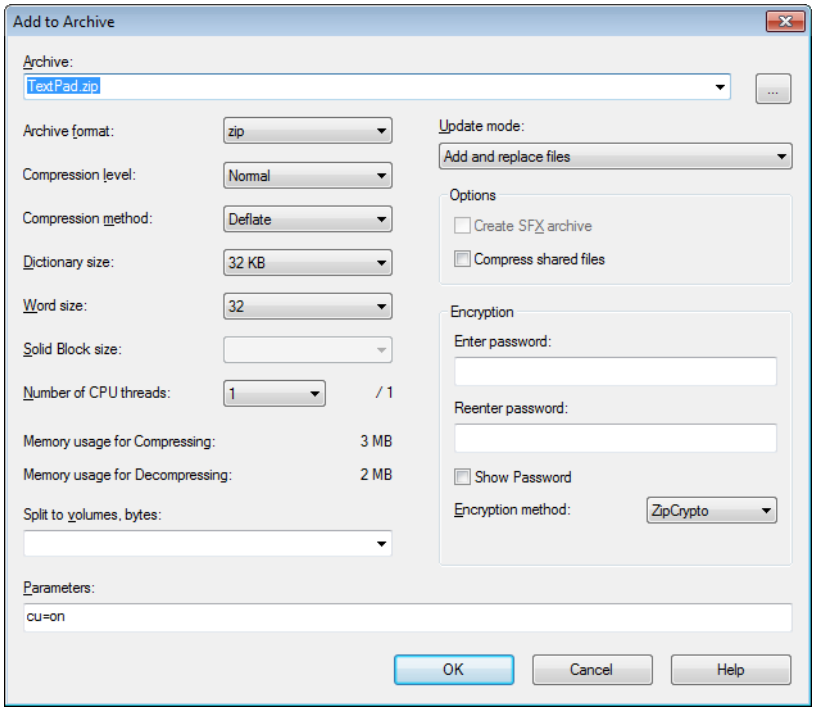


Figure 8-32: The 7-Zip GUI

4. Click **Add**.
5. The 7-Zip Add to Archive dialog appears.



6. Enter “cu=on” into the Parameters text box.
7. Click **OK**.
8. After 7-Zip creates the new ZIP archive, change the extension of the archive file to “stp.”

CAUTION

You can also right click on the appset files from within Windows Explorer and use the context menu to launch 7-zip. In order for 7-zip to create the appset correctly, you must select **Add to Archive**. Do not use any of the other options which bypass the Add to Archive dialog above.

Chapter 9

Troubleshooting

This chapter covers these topics:

- **Identifying Common Problems**
- **Solutions to Common Problems**
  1. Missing Shortcuts
  2. File Associations Do Not Work
  3. Application Fails to Launch
  4. Printing Problems
  5. Fonts Do Not Appear
  6. Folder or Files Disappear from the Virtualized Application
  7. Hide "Add/Remove Programs" Entry
  8. Roaming and Folder Redirection Issues
  9. Prerequisites
  10. Unique User Accounts
  11. Service Show as Running and Cannot Be Stopped
  12. Windows Services or Drivers Do Not Start
  13. Plug and Play Device Drivers
  14. Alternative Data Streams
  15. Anti-virus and Firewall Applications
  16. Application Copy Protection
- **Approaches to Uncommon Problems**

Overview

During the Testing and Optimizing phase, you may face certain complicating issues unique to the application which you are cloudifying. This chapter will describe the most common of these situations and their solutions, along with techniques for dealing with less common problems.

Identifying Common Problems

Occasionally, there are errors that can be difficult to identify. If not all files or registry keys seem to have been captured, use the following troubleshooting steps (if applicable) to identify them:

1. Extract the appset file on a different PC.

   This can be useful for a couple of reasons. It allows you to examine the file structure and registry keys to verify that everything necessary for the program to function is being captured. You can also extract the appset file on a clean PC to verify that the application will run on that computer. You can then verify that all the necessary files, registry keys, environment variables, fonts, and services are included in the appset. For more information, see Extracting an Application for Patching on page 160.

2. Review the Jukebox Player Log file.

   If the extracted appset worked correct, then most likely there is a problem with the settings. Viewing the Jukebox Player log can help to identify what the issues may be. Also check the following:

   - Check that the disposition layer makes the file or key accessible to the application attempting to access it.
   - Check that the security setting allows the file to be found and read.

3. Review the Jukebox Studio Log file.

   This log will contain information on files and registry key captured and filtered from the appset. It will also contain information on any errors that may have been encountered. The Jukebox Studio can be viewed by clicking the **Application** menu and selecting **View Log**. Review the warning given during the capture about security descriptors, user accounts, etc. and will show a listing of files and registry keys not merged.

Solutions to Common Problems

While cloudifying an application will typically have successful results, there are times when the results may not match expectations. For example, did the application create a desktop shortcut and is the expectation that there should be a shortcut? Virtualization layer and security settings can also add to the complexity of how applications should behave for users. Below are solutions to some of the most common problems.

Missing Shortcuts

Most applications create shortcut files (.lnk) on the Desktop or in the Start menu. These files can be located in the Jukebox Studio File view located in the **CommonPrograms** or **UserProgram** folder for the **Start** menu, and located in the **CommonDesktop** or **UserDesktop** folder for the Windows desktop. You may add or remove shortcut files as desired to these folders.

If the shortcut files are missing from Jukebox Player after virtualization, then do the following:

1. In order for the shortcut to appear on the **Start** menu, verify that the shortcut file is located in the **CommonPrograms** or **UserPrograms** folder. In order for the shortcut to appear on the Windows desktop, verify that the shortcut file is located in the **CommonDesktop** or **UserDesktop** folder.
2. Be sure that the Disposition layer for the shortcut files and folders are set to **Integrated** (Disposition layer 3). This will allow shortcut to be accessible on the OS of the Target PC.
3. Check that the security setting on the shortcut files do not have “**Hide from Folder Listing**” selected. If this option is selected, the shortcut files will not be visible on the Target PC.

Blank Shortcuts

If the Desktop icon for the application shows as a blank default Windows icon, the executable file for the application is probably set to “Prevent Copying” in the AIB file. This is normal behavior with this setting and the application will function normally.

File Associations Do Not Work

Windows manages file extension associations to applications under the HKEY_CLASSES_ROOT hive of the Window registry. For file associations to work the file type must be set in the registry and the application executable must be visible on the Target PC. To fix file association issues, do the following:

1. Ensure that the file extension exists in the Jukebox Studio Registry view. If the extension is missing, then it might not have been setup by the application. Either manually add the extensions, or use re-capture the application with “All processes” selected, run the application, and configure the default associations.
2. Be sure that all the extensions are set to **Integrated** (Disposition layer 3) so that they are accessible on the Target system.
3. Change the application main executables to **Integrated** (Disposition layer 3) so that the associated file can be opened by the application.

Application Fails to Launch

Sometimes an application can fail to launch when the command line is incorrect; the command line may have to be specified with a different executable or may need additional launch parameters. Check the original application shortcut, typically found in the Start menu, and use the same command line and parameters in Studio.

Another problem may be that some applications modify the local machine's PATH environment variable during installation or after launches for the first time. There may be some system specific entries included in the PATH variable upon cloudifying. Ensure that it contains no system specific entries. Target systems will have many different configurations and it is important to leave only the entries that are specific to the application being cloudified.

For example: **PATH = %PATH%;C:\ApplicationPath1;C:\ApplicationPath2**

To check if an application populates its own PATH environment variable after capturing an installation, go to the Environment view to check if the environment variable named “PATH” exists.

Printing Problems

Some application will install new printer devices, which can be included in an appset. In the Files view there is a folder named **spool** which is a subdirectory found under the SystemX86Dir system folder. Applications that come with printer drivers may not work properly if this folder is removed from the Files view. Within the Registry view, the registry key **HKEY_LOCAL_MACHINE \ SYSTEM \ CurrentControlSet \ Control \ Print \ Environments \ Windows NT x86 \Drivers** contains references to the printer drivers in the spool folder. If a new printer does not appear, be sure the **Restart Print Spooler** option under the **Virtualization** screen of the **Settings** dialog has been selected.

Fonts Do Not Appear

Most applications come with their own program fonts. If an application is cloudified with fonts set to **Injected** under the **Virtualization** settings, then running the paged application from a shortcut will cause the fonts to not appear. Private fonts are only injected into the paged application if it is launched from Jukebox Player. To make fonts available to all applications, change the **Font** setting to **Register**. In addition to the Fonts view, registered fonts can be found in the following locations:

- In the Files view, the fonts directory can be found at **SystemFonts**
- In the Registry view, the fonts can be found at **HKEY_LOCAL_MACHINE \ SOFTWARE \ Microsoft \ Windows NT \ CurrentVersion \ Fonts**

Folder or Files Disappear from the Virtualized Application

If you happen to cloudify an application to an **install folder** that has the same name as a folder on the local PC, then the local folder may disappear. This can also affect files under common folders marked with disposition layer 3 for integrated, but not merged (see Merged and Non-merged Folders on page 73). To solve this issue, be sure to install the application to a unique install folder name. You may also need to mark files in common folders as isolated if desired. To fix the existing appset:

1. Rename the **Install folder** to be more unique.
2. Set specific files in common locations to be **layer 4** - isolated or merged.

Hide "Add/Remove Programs" Entry

Many application will place an entry in the “Add or Remove Programs” listing of the control panel (or “Programs and Features” on Vista/7). This can be confusing as the application is virtual and cannot be uninstalled. If you wish to hide this entry then take the following steps:

1. Under the Registry view, locate the following key **HKEY_LOCAL_MACHINE \ SOFTWARE \ Microsoft \ Windows \ CurrentVersion \ Uninstall**
2. Remove it from the workspace.

Roaming and Folder Redirection Issues

Some Enterprise deployments may use “Roaming Profiles” and/or “Folder Redirection” to preserve user application settings. For Roaming Profile support, it is important to select **Roam application changes** under the **Sandboxing Settings**. Jukebox Player will then allow HKEY_CURRENT_USER registry settings and the %appdata% folder contents for paged applications to roam properly. Any limitations with Windows Roaming Profiles will still apply.

With Folder Redirection to a network share (via Mapped drive or a UNC path), only Layer-1 (recommended) and/or Layer-2 files are allowed. If any Layer-3 or Layer-4 files are located in these folders, then the appset will fail to activate. For example, if the %appdata% folder is being redirected to \\fileshare\user1\appdata\, then the appset template folder ?roamingappdata? must only contain Layer-1 files and folders. Redirection to another local disk supports virtual files (Layer-3 or Layer-4).

CAUTION

With folder redirection, the network folder must exist prior to activating the application. For example, if %appdata% is mapped to n:\{username}\appdata, then the “appdata” folder must already exist on the server.

Prerequisites

Many applications require the presence of specific runtime library or a minimum version of the .NET Framework. Jukebox Player comes with the latest builds of the following C++ runtime libraries:

- VC++ 2005 x64
- VC++ 2008 x64
- VC++ 2005 x86
- VC++ 2008 x86

Unless you are certain that other prerequisites will be present on the Host PCs, they should be installed and captured prior to capturing the main application. If possible, set the default disposition for new assets to level 4 before capturing. (see Globally Changing Asset Dispositions on page 113 for more information).

It may be necessary to change the security override settings for some processes to keep the assets for a dependency sufficiently isolated.

For example, because Jukebox Player already installs the latest 2005/2008 C++, if the captured application contains these then the process “csrss.exe” should be removed from the Security Override list since it deals with side-by-side assemblies. Also, the disposition for any of the following captured assets should be set to layer-4 as previously mentioned:

<WindowsDir>\winsxs\ all folders and files
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\SideBySide\
HKEY_LOCAL_MACHINE\Software\Classes\Installer\Win32Assemblies\

If an application requires the presence of a specific service pack version of the operating system, you should create a Configurable AppEvent to check for its presence on the Target PC after the application is registered, and alert the user if it is not installed. See About AppEvent Scripts and Executables on page 61 for more information.

Unique User Accounts

Some applications create a special user account when you install them. Jukebox Studio cannot capture user accounts, because these are not transferable between computers. You need to create an AppEvent to programmatically replicate the user accounts and password on the host computer when Jukebox Player virtualizes the application.

NOTE

User Accounts may appear as a security identifier (SID) in the registry. For example, S-1-5-18 is the local system account. Please see http://support.microsoft.com/kb/243330 for more details.

Service Show as Running and Cannot Be Stopped

All executable files that are virtual (layer-3 or layer-4) will be tracked for usage and shown with a “Running” status when being used. This applies to services as well. To not be tracked for usage, the files (i.e. Service ImagePath) would need to be mark for a layer-2 disposition.

In addition, some services may be configured to automatically restart if interrupted. This can be found by running the **services.msc** command, selecting **properties** on the service in question, and going to the **Recovery** view. If the **Subsequent failures** setting is not set to **Take No Action**, then Jukebox Player will not be able to stop the service and remove the application. To correct this it is best to remove the registry value called **FailureActions** under HKEY_LOCAL_MACHINE \ SYSTEM \ CurrentControlSet \ Services in the Register view for that service.

Windows Services or Drivers Do Not Start

In most cases, Jukebox Player handles initialization and starting services and drivers without further intervention. If a service needs to start using a unique user account (this normally defaults to LocalSystem), then the appset will need an AppEvent to handle creating this account.

CAUTION

Please be aware that on some systems services have security descriptors that are enforced more strictly. When adding a new service, it is important to match the security descriptor of the installed version. Use **sc sdshow <service_name>** in command prompt to display security descriptor set for the service. Marking an appset to **require system reboot** after adding a service will typically correct the security descriptors.

If the application has boot or system drivers, then mark all the files and registry keys for those services and/or drivers as Layer 2 (Installed-temporary disposition).

1. Under the Services view, right-click the service to select **Service Key Disposition** and mark it as **Layer 2**.
2. Right-click the same service to select **Image File Properties** and mark it as disposition **Layer 2**.

CAUTION

Marking files for drivers and/or services as layer 2 may causes problems if the appropriate registry keys are also not set as layer 2. Be sure to perform both steps.

3. Exclude all LEGACY registry keys for that service/driver and any specific Enum keys:

   HKEY_LOCAL_MACHINE\System\CurrentControlSet\Enum\Root

   HEKY_LOCAL_MACHINE\ System\CurrentControlSet\Services\{NAME}\Enum
4. In the Advance settings, select **Require system reboot**. If you wish to avoid a reboot, then a CAE may be used to start the driver directly using the command:

   ```
   sc.exe start <name of driver>
   ```

For more information on how services work, please refer to the Microsoft Knowledge Base:

http://support.microsoft.com/kb/251192

NOTE

Typically a system driver cannot be stopped and removed from a system without a reboot. Jukebox Player will properly remove these files from the system but the driver will still be loaded until a reboot occurs.

Plug and Play Device Drivers

Plug and play drivers for devices such as printers are installed in two phases.

- The installer copies the driver files to the hard drive.

- Windows associates the files with the correct device.

The first part is simple for Jukebox Studio, but the second part varies from device to device, and may require additional steps, especially under Windows Vista or later operating systems. Because of the differences, you should cloudify a driver separately for XP and Vista or later. 64-bit version will be different from 32-bit device drivers and also may require additional configuration.

Windows XP

Virtualization of drivers under XP is straightforward. After Jukebox Player virtualizes the driver, connect the device. The Found New Hardware wizard will associate the new driver with the device.

Windows Vista or later

With these versions of Windows, Microsoft introduced what is called the Driver Store, a protected repository for trusted device drivers. These platforms will require a Configurable AppEvent to setup the device driver.

CAUTION

Please do not attempt to set files in the Driver Store to layer 1 or 2. This will cause virtualization to fail.

The utility **pnputil.exe** runs from a command line and its primary purpose is to add drivers to the Driver Store and remove them.The basic syntax for adding drivers using the utility is:

```
pnputil.exe -i -a "C:\<folder>\driver.inf"
```

The best approach to cloudifying a driver for Windows Vista or later:

1. Capture the application and drivers as normal. Exclude the folder <WindowsDir>\inf\.
2. Exclude all virtual files used by the device driver. To know which files are use, first run the command devmgmt.msc that will open the Device Manager window. Find the device driver being captured and double-click it. On the Driver tab, click the Driver Details button and a dialog will show a list of files that should be excluded.
3. For TWAIN devices, mark all files located under <WindowsDir>\SSDriver\ as layer-2.
4. Create a Configurable AppEvent, calling **pnputil.exe** using the syntax shown above. The CAE should use the following settings:
   - The trigger should be set to **After virtualization**.
   - The path should be set to **`%SystemRoot%\system32\pnputil.exe`** and mark for **Use local application**.
   - Set the handler launching to **Run elevated**.
5. The driver to register will be normally found under c:\windows\system32\DriverStore\FileRepository as an INF file. Register all the INF files.
6. You will need to alert the user that, when the CAE runs the **pnputil.exe**, Windows will display a security warning. They will need to click the button to allow the installation.

NOTE

Device drivers will remain on the system even after the application has been removed by Jukebox Player. This is expected with device drivers.

Alternative Data Streams

Users may receive warning messages about Alternate Data Streams (ADS) when Jukebox Player virtualizes an application. In most cases, these messages are not a serious problem and may be ignored. In rare cases, the application may require the existence of an ADS for a particular asset. In this case, the appset should exclude the asset from the sandbox. Set the disposition of the asset to level 2 or lower.

Anti-virus and Firewall Applications

While it is possible to capture and create appsets for various anti-virus and firewall applications, please be aware that these applications include drivers and may limit how Jukebox Player works. For example, if the application is a Firewall, it may attempt to block Jukebox Player from properly fetching data over the internet by default. Be sure to add Jukebox Player components to any exceptions rules (e.g. Add streamingcore.exe to a firewall allow list) when the appset is created.

Application Copy Protection

Many applications employ some form of copy protection to prevent theft and enforce proper licensing. Application Jukebox does not attempt to remove such protection. It is ideal to cloudify an application using a serial number that is based for high volume licensing (e.g. Microsoft products use Volume License Keys) so that multiple systems can activate with one key. Otherwise, the application will prompt for licensing as it would normally and/or it may not be possible for every system to activate the application and to make it fully functional.

It is best to test the appset on other systems to ensure there is no form of license enforcement that you are not already aware of. If the appset does not work or prompts for licensing, then retest on an image, or snapshot, of the same system used to cloudify. If the application binds to hardware, then it should work on the same system. Please contact the software ISV for licensing models that they support.

Approaches to Uncommon Problems

If you have identified a problem that you could not solve with the above procedures, use the following guidelines to resolve the problem:

- Cloudify the application again with the “All processes” option.

  These are typically due to files or registry keys that are either not captured by Jukebox Studio or have been removed from the appset. If you suspect that not all files or registry keys were captured, it is best to capture the application again and specify the “All processes” option. If the problem persists, then additional troubleshooting will be required.
- Cloudify the application again without converting short path names to long names.

  A few applications still require file references by the old DOS-style file and folder names. If these files are converted during the merge process, it could break the cloudified application. Capture the application, but uncheck **Replace short path names with long ones in the registry**.
- Open the Event Viewer and review the Application Logs.

  Run Event Viewer (**Windows Control Panel** > **Administrative Tools**) on the Target PC and view the Application Logs and look a detailed description of Errors or Warnings. Many times, the logs will provide information about a file or registry key that cannot be located or accessed.
- Use Process Monitor to view files, registry, and process information.

  Process Monitor, a utility that is available from Microsoft’s website, monitors files, registry, and process information of the application. Process Monitor will display a large amount of system

information, so start by excluding all processes other than the paged application. Next, launch the paged application and search for “NOT FOUND” results. These will indicate that the application was not able to locate a file or registry key.

- Dump the file and registry lists from the AIB file editor.

  Compare these text files to similar outputs from known good appsets for the same or a similar application. Any differences may help to highlight the problem. See Modifying File Settings on page 119 and Modifying Registry Settings on page 122 for more information.

Chapter 10

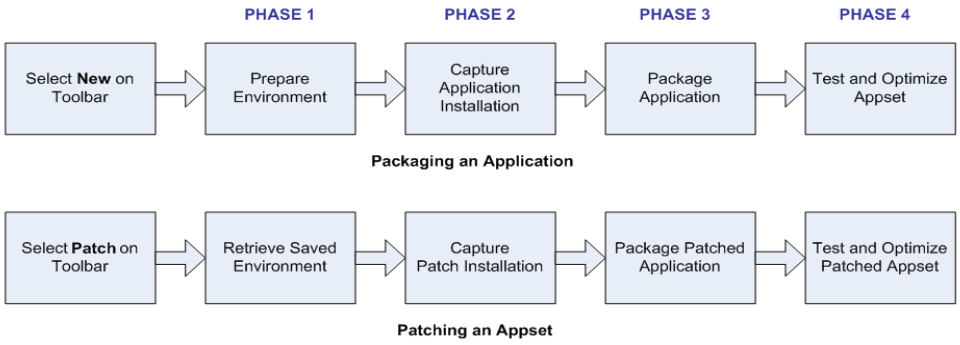
Patching an Appset

This chapter covers these topics:

- **Overview**
- **Patching an Appset**
- **Modifying a Patched Appset**
- **Extracting an Application for Patching**

Overview

Periodically, applications require program patches, which must be incorporated into the existing appset files. Patch information is captured by running the patch provided by the software vendor, and capturing the changes in the files and registry settings. These changes are stored in the appset file (**.stp**), and are utilized to create an updated content file (STC), metadata file (.AIB), and token file (**.tok**).



Patching an Appset

Patching allows application updates to be captured and recorded in the appset. Appsets can contain multiple patches, which are added cumulatively. Each patch adds the required new files and registry settings to the STC file, and creates a new AIB file as well.

The following is required in order to create a patched appset:

- The environment that was used to create the current appset (copy of VM or image of physical PC)
- The Jukebox Studio project file (**.stw**) for the appset being patched.
- The appset file (**.stp**).

If you don't have access to the original environment, you can recreate the project file on a new machine by unpacking the appset file (**.stp**). For more information, see Extracting an Application for Patching on page 160.

CAUTION

Make sure you have opened the correct project file before patching. And make sure that only one *.stw file is open, including the "**Untitled*.stw**" file.

To patch an existing appset:

1. Either restore the base environment that was used to create the appset from a VM or a physical PC image, or extract the appset (.stp) file to a clean PC.
2. Launch Jukebox Studio.
3. Click **Patch** on the Jukebox Studio toolbar, navigate to the project file (**.stw**) for the appset, and click **Open**.
4. Enter a descriptive name in the Patch Title box. This could include the version and any other information that would enable users to be sure they had the latest version.

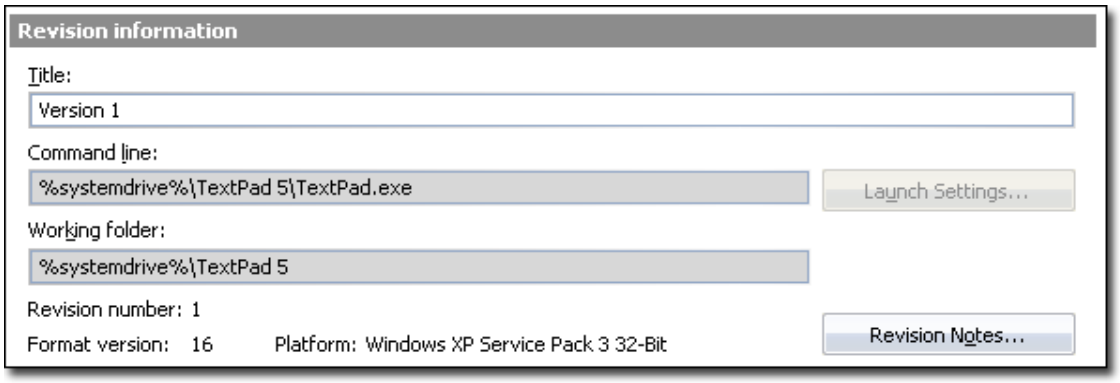


Figure 10-1: Patch Information

The launch settings button will be disabled, since you cannot change the command line for an existing application.

5. If desired, enter new revision notes for this patch. Click the **Revision Notes...** button to display the **Revision Notes** dialog. See Revision Notes on page 40 for more information.

6. Click **Capture** on the Jukebox Studio toolbar. *Alternative*: Click **Application** on the menu bar, then select **Capture Application**. The **Capture Application Installation** dialog appears.

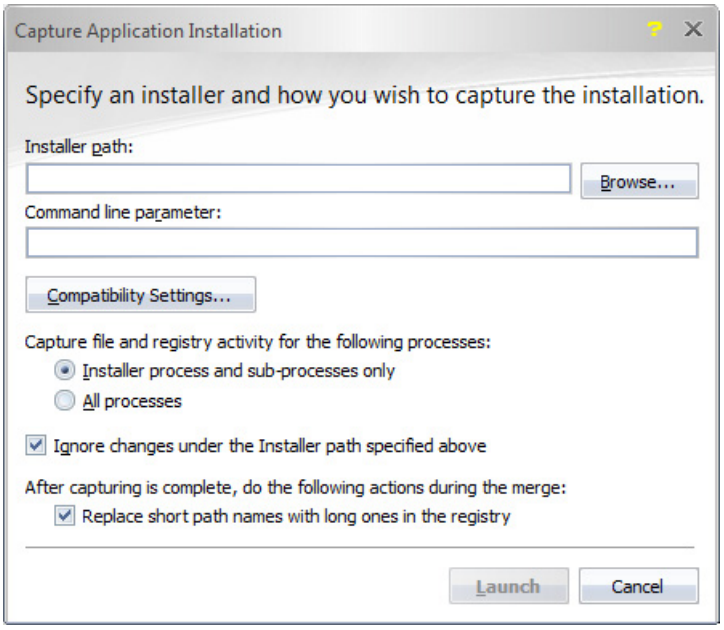

Capture Application Installation
Specify an installer and how you wish to capture the installation.
Installer path:
Browse...
Command line parameter:
Compatibility Settings...
Capture file and registry activity for the following processes:
Installer process and sub-processes only
All processes
Ignore changes under the Installer path specified above
After capturing is complete, do the following actions during the merge:
Replace short path names with long ones in the registry
Launch
Cancel

Figure 10-2: Capture Application Installation Dialog

7. Click **Browse** and locate the software vendor's patch installer (.EXE) for the application.
8. Click **Open**, and then click **Launch**.
9. Select the destination folder, then follow the vendor's patch installation instructions to completion.

   For information about processes or compatibility settings, see Start the Application Installation Capture Process on page 33.

   The Files and Registry views will show which assets have been added, modified, or removed from the project by the new patch.

   - Assets shown in **blue** are new. Installing the patch added them to project.
   - Assets shown in **magenta** have been modified. Installing the patch overwrote the existing files with newer versions.
   - Assets shown in **gray** were removed from the project.
10. Save the patched project file (**.stw**).
11. See Cloudifying the Application on page 43 for cloudifying instructions.
12. See Testing and Optimizing the Appset on page 47 for instructions on testing and optimizing the patched appset.

    The patched appset is now ready to publish to Jukebox Server.

To add a prefetch to a patch appset:

When you create a patch, any prefetch files that have not changed from the previous version of the application will be carried over to the prefetch for the patch. Files that have changed will be removed. Please follow the steps under **Chapter 5**, Testing and Optimizing the Appset on page 47 for adding the new prefetch.

NOTE

Once you patch an appset, you can no longer modify the prefetch of the previous versions. You can only add or remove files from the prefetch of previous versions by editing the AIB files that correspond to those versions.

Modifying a Patched Appset

For instructions about modifying a patched appset, refer to **Chapter 7**, Modifying Project Assets on page 71.

Extracting an Application for Patching

In previous versions of Jukebox Studio, in order to patch an appset, you had preserve the Windows environment upon which the original application was installed. This normally meant backing up the virtual machine or saving an image of the hard disk using a utility like Symantec's *Ghost*.

Jukebox Studio can now extract assets of the original application from the appset, effectively recreating the original installation even if you don't have access to the original computer. Along with the original STW project file, you can then follow the steps above for patching an appset.

Extracting is also useful to examine or troubleshoot an appset. By extracting the appset to a new system, the application can be tested as a normal installation to ensure the appset contains the correct information.

To extract an application from an appset:

1. Start with a freshly installed Windows environment, running the same Windows version that was in use when the appset was originally created.
2. Copy the appset (**.stp**) file to a folder on this system.
3. Open Jukebox Studio.
4. Click the **Application** menu and select **Extract Application...**

The **Extract Application** dialog appears.

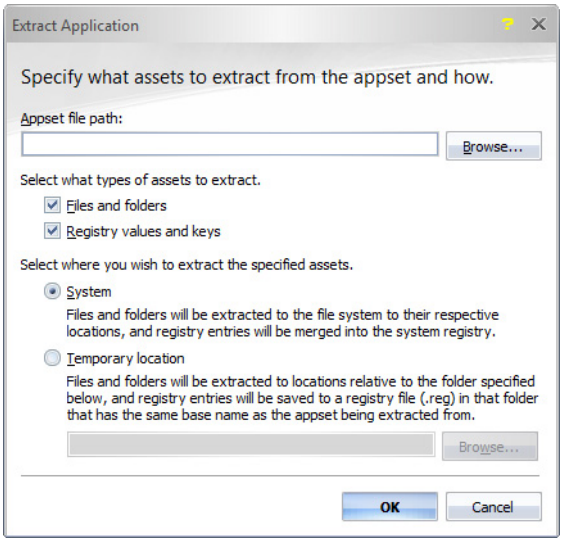

Extract Application
Specify what assets to extract from the appset and how.
Appset file path:
Browse...
Select what types of assets to extract.
Files and folders
Registry values and keys
Select where you wish to extract the specified assets.
System
Files and folders will be extracted to the file system to their respective locations, and registry entries will be merged into the system registry.
Temporary location
Files and folders will be extracted to locations relative to the folder specified below, and registry entries will be saved to a registry file (.reg) in that folder that has the same base name as the appset being extracted from.
Browse...
OK
Cancel

Figure 10-3: The Extract Application Dialog

5. Click **Browse** to locate the appset (**.stp**) file to be extracted.
6. Under Extraction, you can uncheck **File** or **Registry** if you do not need these parts of the appset extracted into the new project.

NOTE

In order to patch an appset, you must extract both file and registry assets when extracting. You can choose not to extract one or the other if you are examining or troubleshooting an appset.

7. Select the **Location** where Jukebox Studio will place the files for the extracted application.
   - **System** – Files are extracted to their original location (where they were stored on the PC when the appset was originally cloudified).
   - **Temporary** – Files are extracted to folder, which you specific by clicking the **Browse** button.
8. Click **OK** to begin extracting the application.

NOTE

Extraction does not run the Configurable AppEvents. After extraction, any commands added as a CAE can be launched manually, if necessary.

When Jukebox Studio is finished extracting the application, you can then open the original project file (STW), apply the patch, and re-cloudify the appset as described above.

Appendix

A

Folder Template Mapping

Folder Name	Template	Default Path - XP	Default Path - Vista or Later
AppData	?roamingappdata?	C:\Documents and Settings\<username>\Application Data	C:\Users\<username>\AppData\Roaming\
CommonAdminTools	?commonadmintools?	C:\Documents and Settings\All Users\Start Menu\Programs\Administrative Tools	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Administrative Tools
CommonDesktop	?publicdesktop?	C:\Documents and Settings\All Users\Desktop	C:\Users\Public\Desktop
CommonDocuments	?publicdocuments?	C:\Documents and Settings\All Users\Documents	C:\Users\Public\Documents
CommonDownloads	?publicdownloads?	N/A*	C:\Users\Public\Downloads
CommonGamesTasks	?publicgametasks?	N/A*	C:\ProgramData\Microsoft\Windows\GameExplorer
CommonMusic	?publicmusic?	C:\Documents and Settings\All Users\Documents\My Music	C:\Users\Public\Music
CommonOEMLinks	?commonoemlinks?	C:\Documents and Settings\All Users\OEM Links	C:\ProgramData\OEM Links
CommonPictures	?publicpictures?	C:\Documents and Settings\All Users\Documents\My Pictures	C:\Users\Public\Pictures
CommonPrograms	?commonprograms?	C:\Documents and Settings\All Users\Start Menu\Programs	C:\ProgramData\Microsoft\Windows\Start Menu\Programs
CommonProgramFiles	?programfilescommon?	C:\Program Files\Common Files	C:\Program Files\Common Files
CommonProgramFilesX64	?programfilescommonx64?	See Remarks Table (For 32-bit and 64-bit Folder Templates) on page 166	See remark
CommonProgramFilesX86	?programfilescommonx86?	See Remarks Table (For 32-bit and 64-bit Folder Templates) on page 166	See remark
CommonSidebarDefaultParts	?sidebardefaultparts?	N/A*	C:\Program Files\Windows Sidebar\Gadgets
CommonStartMenu	?commonstartmenu?	C:\Documents and Settings\All Users\Start Menu\Programs	C:\ProgramData\Microsoft\Windows\Start Menu\Programs
CommonStartup	?commonstartup?	C:\Documents and Settings\All Users\Start Menu\Programs\StartUp	C:\ProgramData\Microsoft\Windows\Start Menu\Programs\StartUp

Folder Name	Template	Default Path - XP	Default Path - Vista or Later
CommonTemplates	?commontemplates?	C:\Documents and Settings\All Users\Templates	C:\ProgramData\Templates
CommonVideos	?publicvideos?	C:\Documents and Settings\All Users\Documents\My Videos	C:\Users\Public\Videos
LocalAppData	?localappdata?	C:\Documents and Settings\<username>\Local Settings\Application Data	C:\Users\<username>\AppData\Local\
LocalizedResourcesDir	?localizedresourcesdir?	C:\Windows\resources\0409 (code page)	C:\Windows\resources\0409 (code page)
ProgramData	?programdata?	C:\Documents and Settings\All Users\Application Data	C:\ProgramData
ProgramFilesX64	?programfilesx64?	See Remarks Table (For 32-bit and 64-bit Folder Templates) on page 166	See remark
ProgramFilesX86	?programfilesx86?	See Remarks Table (For 32-bit and 64-bit Folder Templates) on page 166	See remark
PublicDir	?public?	N/A*	C:\Users\Public
ResourceDir	?resourcedir?	C:\Windows\Resources	C:\Windows\Resources
SampleMusic	?samplemusic?	C:\Documents and Settings\All Users\Documents\My Music\Sample Music	C:\Users\Public\Music\Sample Music
SamplePictures	?samplepictures?	C:\Documents and Settings\All Users\Documents\My Pictures\Sample Pictures	C:\Users\Public\Pictures\Sample Pictures
SamplePlaylists	?sampleplaylists?	N/A*	C:\Users\Public\Music\Sample Playlists
SampleVideos	?samplevideos?	N/A*	C:\Users\Public\Videos\Sample Videos
SystemDir	?system?	C:\Windows\System32	C:\Windows\System32
SystemFonts	?fonts?	C:\Windows\Fonts	C:\Windows\Fonts
SystemX86Dir	?systemx86?	See Remarks Table (For 32-bit and 64-bit Folder Templates) on page 166	See remark
UserAdminTools	?admintools?	C:\Documents and Settings\<username>\Start Menu\Programs\Administrative Tools	C:\Users\<username>\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Administrative Tools
UserCDBurning	?cdburning?	C:\Documents and Settings\<username>\Local Settings\Application Data\Microsoft\CD Burning	C:\Users\<username>\AppData\Local\Microsoft\Windows\Burn\Burn
UserContacts	?contacts?	N/A*	C:\Users\<username>\Contacts
UserCookies	?cookies?	C:\Documents and Settings\<username>\Cookies	C:\Users\<username>\AppData\Roaming\Microsoft\Windows\Cookies

Folder Name	Template	Default Path - XP	Default Path - Vista or Later
UserDesktop	?desktop?	C:\Documents and Settings\<username>\Desktop	C:\Users\<username>\Desktop
UserDocuments	?documents?	C:\Documents and Settings\<username>\My Documents	C:\Users\<username>\ Documents
UserDownloads	?downloads?	N/A*	C:\Users\<username>\Downloads
UserFavorites	?favorites?	C:\Documents and Settings\<username>\Favorites	C:\Users\<username>\Favorites
UserGameTasks	?gametasks?	N/A*	C:\Users\<username>\AppData\Local \Microsoft\Windows\GameExplorer
UserHistory	?history?	C:\Documents and Settings\<username>\Local Settings\History	C:\Users\<username>\AppData\Local\ Microsoft\Windows\History
UserInternetCache	?internetcache?	C:\Documents and Settings\<username>\Local Settings\Temporary Internet Files	C:\Users\<username>\AppData\Local\ Microsoft\Windows\Temporary Internet Files
UserLinksDir	?links?	N/A*	C:\Users\<username>\Links
UserLocalAppDataLow	?localappdatalow?	N/A*	C:\Users\<username>\AppData\LocalLow\
UserMusic	?music?	C:\Documents and Settings\<username>\My Documents\My Music	C:\Users\<username>\Music
UserNetworkShortcuts	?nethood?	C:\Documents and Settings\<username>\NetHood	C:\Users\<username>\AppData\Roaming\Microsoft\Windows\Network Shortcuts
UserOriginalImages	?originalimages?	N/A*	C:\Users\<username>\AppData\Local\Microsoft\Windows Photo Gallery\Original Images
UserPhotoAlbums	?photoalbums?	N/A*	C:\Users\<username>\Pictures\Slide Shows
UserPictures	?pictures?	C:\Documents and Settings\<username>\My Documents\My Pictures?	C:\Users\<username>\Pictures
UserPrinterShortcuts	?printhood?	C:\Documents and Settings\<username>\PrintHood	C:\Users\<username>\AppData\Roaming\Microsoft\Windows\Printer Shortcuts
UserProfile	?profile?	C:\Documents and Settings\<username>	C:\Users\<username>
UserProfiles	?userprofiles?	N/A*	C:\Users
UserPrograms	?programs?	C:\Documents and Settings\<username>\Start Menu\Programs	C:\Users\<username>\AppData\Roaming\Microsoft\Windows\Start Menu\Programs
UserQuickLaunch	?quicklaunch?	C:\Documents and Settings\All Users\Application Data\Microsoft\Internet Explorer\Quick Launch	C:\Users\<username>\AppData\Roaming\Microsoft\Internet Explorer\Quick Launch
UserRecent	?recent?	C:\Documents and Settings\<username>\Recent	C:t\Users\<username>\AppData\Roaming\Microsoft\Windows\Recen

Folder Name	Template	Default Path - XP	Default Path - Vista or Later
UserSavedGames	?savedgames?	N/A*	C:\Users\<username>\Saved Games
UserSearches	?savedsearches?	N/A*	C:\Users\<username>\Saved Searches
UserSendTo	?sendto?	C:\Documents and Settings\<username>\SendTo	C:\Users\<username>\AppData\Roaming\Microsoft\Windows\SendTo
UserStartMenu	?startmenu?	C:\Documents and Settings\<username>\Start Menu	C:\Users\<username>\AppData\Roaming\Microsoft\Windows\Start Menu\Programs
UserStartup	?startup?	C:\Documents and Settings\<username>\Start Menu\StartUp	C:\Users\<username>\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup
UserTemplates	?templates?	C:\Documents and Settings\<username>\Templates	C:\Users\<username>\AppData\Roaming\Microsoft\Windows\Templates
UserVideos	?videos?	C:\Documents and Settings\<username>\My Documents\My Videos	C:\Users\<username>\Videos
WindowsDir	?windows?	C:\Windows	C:\Windows

* N/A means not supported on the platform

Table A-1: Remarks Table (For 32-bit and 64-bit Folder Templates)

Directory Template	Operating System	Default Path
?programfilesx86?	32 bit	C:\Program Files
	64 bit	C:\Program Files (x86)
?programfilesx64?	32 bit	N/A*
	64 bit	C:\Program Files
?programfilescommonx86?	32 bit	C:\Program Files\Common Files
	64 bit	C:\Program Files (x86)\Common Files
?programfilescommonx64?	32 bit	N/A*
	64 bit	C:\Program Files\Common Files
?system?	32 bit	C:\Windows\System32
	64 bit	C:\Windows\System32
?systemx86?	32 bit	C:\Windows\System32
	64 bit	C:\Windows\Syswow64

* N/A means not supported on the platform

Appendix

B

Editing Configuration Files

The files detailed in this appendix control some of the default behaviors of Jukebox Studio or default settings for the cloudified application. Editing these files allow you to control which elements of an application are captured and how the final appset interacts with the host PC where AJP is installed. These files are installed into the \lib sub-folder of the main Jukebox Studio, so the standard path to this files are:

```
[Install Path]\lib
```

There are eleven .Dat files in this folder. These are all text files, which can be edited with Windows Notepad or another text editor.

Conventions

Comment lines begin with a "#" character. Any non-blank line that doesn't begin with this character is considered a separate, active item and Jukebox Studio will attempt to act up on the information contained in that line.

Each file contains default entries designed to work in most situations. These also function as examples for adding new entries.

preloadreghives.dat

This file defines registry hives that must be loaded prior to capturing. Each entry specifies a hive to be loaded into the system registry using a RegLoadKey call.

Hives that are normally loaded by system at boot time (e.g. HKEY_LOCAL_MACHINE\Software) need not be listed in this file.

Example

```
HKEY_LOCAL_MACHINE\COMPONENTS    %windir%\System32\Config\Components
```

procexcluded.dat

This file defines processes that will be ignored during Capture, even when the capture is launched with the "Capture All Processes" option checked.

The structure of each entry consists of the absolute path to the process executable to be excluded.

```
C:\WINDOWS\SYSTEM32\SEARCHINDEXER.EXE
```

filefilt.dat

This file defines files and folders that will be filtered out of the project file during the capture and merge process.

Each entry in this file consists of NAME followed by the path.

Example:

```
USERDATA_1 %userprofile%\NTuser.dat
```

NOTE

This file is loaded once every time a new project is created. Subsequently, loaded information from this file will be saved with the new project. Reopening an existing project will not reload this file and information will be reloaded from the saved project. In other words, changes to this file will affect only future projects when they are created.

regfilt.dat

This defines registry paths to which changes are excluded from the capture and merge process.

Each entry consists of a filter name (FILTER_001, FILTER_002), followed by white space and the full registry path of the key to be filtered.

NOTE

This file is loaded once every time a new project is created. Subsequently, loaded information from this file will be saved with the new project. Reopening an existing project will not reload this file and information will be reloaded from the saved project. In other words, changes to this file will affect only future projects when they are created.

procfilt.dat

This file defines which processes captured by Application Jukebox should be included when the captured installed is merged into the project file. If this file is blank, all processes are merged. If any entries are found, only those processes are captured and merged.

Each entry consists of the process executable name (SPOOLSV.EXE), followed by a TRUE (all new child processes spawned by the process will be captured and merged) or FALSE (all child processes spawned by the process will be ignored). Existing child processes are always ignored.

defprocsel.dat

This file controls the default settings for the Security screen of the Options dialog (see Security Settings on page 66). The process names and their settings in this file will be the initial settings for a new project.

Each entry in this file consists of the executable name for the process, some white space, TRUE or FALSE, more white space, and if necessary, TRUE or FALSE again. The TRUE/FALSE settings determine the initial security settings for the listed process.

The first TRUE or FALSE defines the SELECT-ability of level 4 files: can a process on a host PC detect the existence of file within the virtual namespace. The section TRUE or FALSE defines the readability: can the process actually access the contents of the file.

Example:

```
PROCESS1.EXE        TRUE    TRUE
PROCESS2.EXE        TRUE    FALSE
PROCESS3.EXE        FALSE
```

The first entry defines a process that can detect and read files in the virtualized appset, even if their disposition is virtual-isolated (level 4).

The second entry defines a process that can detect virtual files, even if their disposition is virtual-isolated (level 4), but not read them.

The third entry defines a process that cannot detect files in the virtualized appset, even if their disposition is virtual-integrated (level 3).

If the first value is FALSE, any second value will be ignored if present, because if a file cannot be selected, by definition it cannot be detected.

NOTE

This file is loaded once every time a new project is created. Subsequently, loaded information from this file will be saved with the new project. Reopening an existing project will not reload this file and information will be reloaded from the saved project. In other words, changes to this file will affect only future projects when they are created.

fileexcluded.dat

This configuration file defines which files and folders are excluded from Application Jukebox Player's normal sandboxing behavior (see Sandboxing Settings on page 63). The file paths listed here on the host PC will be accessible to the virtualized application, and changes would be written to the physical folders, rather than be sandboxed within Jukebox Player's virtual namespace.

Examples:

An absolute path: `C:\Abc\Xyz`
Special folder templates: `?documents?` for *My documents*
Environment variables: `%programfiles%\xyz`

By default, the personal document folders ("My Documents," "My Pictures," etc.) are excluded from sandboxing.

NOTE

This file is loaded once every time a new project is created. Subsequently, loaded information from this file will be saved with the new project. Reopening an existing project will not reload this file and information will be reloaded from the saved project. In other words, changes to this file will affect only future projects when they are created.

regexcluded.dat

This configuration file defines which registry keys are excluded from Application Jukebox Player's normal sandboxing behavior. (see Sandboxing Settings on page 63). The registry paths listed here on the host PC will be accessible to the virtualized application, and changes would be written to the local registry, rather than be sandboxed within Jukebox Player's virtual namespace.

By default, this file is empty and all registry keys are sandboxed.

NOTE

This file is loaded once every time a new project is created. Subsequently, loaded information from this file will be saved with the new project. Reopening an existing project will not reload this file and information will be reloaded from the saved project. In other words, changes to this file will affect only future projects when they are created.

regdifferences.dat

This file defines registry keys and registry values which will be compared before and after capturing the installation, keeping the difference between the original value and the new value. For example, if the original value was "abc" and after installation the value is "abcdef," only the "def" portion is written to the registry when virtualizing the application on a host PC.

Separate the registry key path and value name using a tab. If key path contains spaces, then enclose the path in quotes (") and escape any backslash characters (\\).

rlfnfilt.dat

This file defines registry key where the data is path names is stored in the registry as a short file name, which will be converted to the full long file name in the appset file.

For example, a folder named "Microsoft Office" might be shown as "MICROS~1" in a registry key. This might cause Player to confused this folder with another folder on the Target PC, which may have assigned "MICROS~1" to a different long folder name, such as "Microsoft Silverlight."

File paths in the registry keys listed in his file will be converted to their long file name (LFN) equivalent.

appratng.dat

This file is used to define the local game ratings. By default, this file contains entries relevant to the United States. If using Jukebox Studio with games from another locality, you can edit this file to localize Jukebox Studio to the rating system in use in that country.

NOTE

This file is loaded once every time a new project is created. Subsequently, loaded information from this file will be saved with the new project. Reopening an existing project will not reload this file and information will be reloaded from the saved project. In other words, changes to this file will affect only future projects when they are created.

Appendix C

Command Line Usage

Jukebox Studio can be used from the command line according to the following syntax:

```
JukeboxStudio [file_path | {-d aib_file -f file_dump -e[:r]
   registry_dump} | {{-p workspace_file} -o output_file [-i [app_id]]
   PREFETCH_OPTIONS}] [-l log_file_override] [-h | -?] PREFETCH_OPTIONS:
   [-m[:n] prefetch_file [...]] [-r[:n] prefetch_file [...]]
```

The various command line parameters are described in the table below:

`file_path`	Open specified .stw, .tok or .aib file in Application Jukebox Studio
`-p workspace_file`	Create appset specified by workspace (.stw) file. If option -c is also specified, -p is ignored.
`-d aib_file`	Dump file and registry entries from the specified aib (.aib) file.
`-f file_dump`	Write file dump to specified file.
`-e[:r] registry_dump`	Write registry entry dump to specified file, resolve templates when optional parameter is specified.
`-o output_file`	Store converted or created appset in specified .stp file.
`-i app_id`	Create appset using a new application ID. Use app_id if specified (a GUID in the form 12345678-90ab-cdef-1234-567890abcdef, with or without enclosing braces); otherwise generate a new application ID automatically.
`-m[:n] prefetch_file`	Merge content of specified file with stage-n prefetch in converted or created appset. If stage number n is not specified, 1 is assumed.
`-r[:n] prefetch_file`	Replace stage-n prefetch in converted or created appset with content of the specified file. If stage number n is not specified, 1 is assumed.
`-l log_file_override`	Write log entries to specified file (overrides the use of the default log file).
`-h, -?`	Display command line help text.

Notes:

- Items in braces {} are to be considered collectively as a group.
- Items or groups in square brackets [] are optional.
- An ellipsis in square brackets [...] indicates that the previous option may be repeated any number of times.
- Pipe characters (|) separate items or groups only one of which may be chosen.

- Forward slashes (/) may be used instead of dashes (-) as a switch prefix.
- Both -r and -m options may be specified simultaneously for the same stage number. In this case the prefetch is first replaced according to the -r option, then augmented according to the -m option.
- - Options -r or -m may be specified multiple times for different stage numbers. If any of these options is specified more than once for the same stage number, only the last option specified for that stage number will apply.

Glossary

The definitions in this glossary are in the context of Application Jukebox Studio and may not necessarily correspond to more general definitions.

access token	A file that is used to monitor the license agreement of a cloudified application, which is passed back and forth between Jukebox Player and the ASP's Jukebox Server. When Jukebox Player receives an access token, permission is granted to run the application. If a user is running multiple or expired **sessions,** Jukebox Player may be denied an access token.
activation	The first of six stages that launch an application from Jukebox Player.
admin	See system administrator.
Admin service	The Jukebox Server component that provides configurable web-based interfaces for the administrator, as well as web interfaces for users to launch cloudified applications.
AIB file	This file contains all the files' metadata and registry information for the cloudified application. The extension for this file is **.aib**.
AppEvent	A Jukebox Studio process or action that uses a **trigger** to run an executable program or script. In turn, the script contains instructions to perform one ore more specified runtime actions that are required by the application being cloudified, such as launching another application. AppEvent scripts and executable programs can be written in any scripting or programming language. AppEvents are also called **Configurable AppEvents** or **CAE**.
application	Any Windows software program or game, such as Microsoft Word or Adobe Acrobat. See also cloudified application.
application license	A license that controls the conditions under which users can access and use an application. These conditions are defined by settings such as license type, total number seats, offline duration (fixed licenses only), etc.
Application Service Provider	(ASP) A provider of ***cloudified applications***, which could be a user's company IT group, a software vendor, or an online gaming company. An ASP packages software applications into a format that can be paged to ***clients*** over the Internet or a company intranet.
appset	See cloudified application.

appset designer The person responsible for using Jukebox Studio to convert Windows-based applications into a cloudified format.

artifacts For Jukebox Studio, artifacts are unwanted changes or additions, such as DLLs, that occur most often when “All processes” is selected during the application installation capture portion of the cloudifying process.

ASP See Application Service Provider.

asset An item added to a project that is needed to cloudify an application. The basic asset types ar files, folders, registry values, registry keys, and environment variables. Other asset types such as fonts, startup items, services, and drivers are themselves made up of one or more file and registry assets.

authentication The process by which the system validates a user's logon information. The user's name and password are compared against an authorized list, and if the system identifies a match, access is granted to the extent specified in the permission list for the user.

buffering See prefetching.

cache (Pronounced “cash”) A temporary storage place (often a specified portion of computer disk space or RAM memory) for frequently needed data that can be retrieved quickly.

See also Player cache.

CAE file (**C**onfigurable **A**pp**E**vents file) - There is one CAE file per AppEvent ***trigger*** which generates a response, such as running a VB script that installs other needed applications.

See also AppEvent.

CGI script A script that uses the CGI (**C**ommon **G**ateway **I**nterface) **protocol** for interfacing external application software with an information server, commonly a ***web server***. In the case of a web server, the CGI script responds to requests from client web browsers by returning output. Each time a request is received, the server analyzes what the request asks for, and returns the appropriate output.

clean cloudifying PC For Jukebox Studio, a PC that has the Windows XP operating system or later installed, as well as the most recent drivers (and possibly imaging software), but is free of any other unnecessary applications.

client For Jukebox Studio, a PC with Jukebox Player installed.

In general, a client is a computer that accesses shared network resources provided by another computer, called a server. Also, a client can be an application or process that requests a service from a process or component. In a client-server environment, the workstation is usually the client computer.

cloudified application An application that has been converted to a cloudified format using Jukebox Studio, that a user can access from any Windows XP or later computer with an Internet broadband or company intranet connection using Application Jukebox Player, without installing the program on a client PC. There are three different names for a "cloudified application" depending on the point of view (POV) of the person using it:

- From an ***Appset Designer***'s POV - "***appset***"
- From a Jukebox Server admin's POV - "***application***"
- From a user's POV - "***cloudified application***"

cloudifying The process of converting a software application into a form that can be paged to users' PCs.

See also cloudified application.

cloudpaging The process of transferring data to a user's computer from an ***ASP***. "Paging" refers to the delivery method of the medium rather than to the medium itself. This distinction is usually applied to media or data that is distributed over the Internet. Application Jukebox "pages" a cloudified application from an ASP to users' PCs.

Configurable AppEvent See AppEvent.

database server A machine that stores profile, usage, and system information of Jukebox Player users. The database server is a component of the Jukebox Server system.

See also Jukebox database.

disposition layers The layers used to configure individual system resources, such as a file, folder, registry key, or registry value. The layer determines whether the system resource can be seen by the local system, and whether it is permanent or can be removed.

Layer 1: Installed-permanent - Copies assets (files, folder, registry keys, and registry values) permanently onto the local system, and can be seen by the entire local system.

Layer 2: Installed-temporary - Installs assets during the activation process, and uninstalls assets during the deactivation process. The original asset is backed up before the new asset is installed, and when the new asset is uninstalled the original asset is restored.

Layer 3: Virtual-integrated - (default setting) Assets that can be seen both by the paged application and the local system, but are not physically installed on the local system.

Layer 4: Virtual-isolated - Assets that can only be seen by the paged application, and are not physically installed on the local system.

See also virtualization.

DSN file (**D**ata **S**ource **N**ame file) This file is used in the process of linking a Jukebox database to a Microsoft Access database.

End User License Agreement (EULA) This is typically an HTML file that consists of a legal agreement the user acknowledges regarding the warranties and conditions of use for a software program.

EULA See End User License Agreement.

failover The process of taking resources offline for one ***node*** and bringing them back online on another node. For example, when one individual computer fails, another machine automatically takes over its request load. The transition is invisible to the user.

See also node and load balancing.

fileshare A folder and its contents that is shared with other users on a network.

firewall A firewall is a software and/or hardware barrier that protects private and company information from external threats. For companies, all communication is routed through a proxy server outside of the organization's network, and the proxy server decides whether it is safe to let a particular message or file pass. The proxy server hides the true network addresses.

fixed license An application license that permits access to the application when the user is working offline (not connected to the ASP's network). NOTE: a fixed license is in use when an application is in the Ready or Running state.

See also floating license.

floating license An application license that requires the user's PC to be online (connected to the ASP's network) in order to access the application. With this license the application cannot be accessed offline. NOTE: A floating license is only in use when an application is Running, but not when it is in the Ready state.

See also fixed license.

FQDN See Fully Qualified Domain Name.

Fully Qualified Domain Name (FQDN) An unambiguous domain name that specifies the exact location in the Domain Name System's tree hierarchy through to a top-level domain, and finally to the root name server. Some applications, such as web browsers, will try to qualify the domain name part of a Uniform Resource Locator (URL) if the DNS resolver can not find the domain. An FQDN differs from a regular domain name by its absoluteness; a default domain name will not be added.

Globally Unique Identifier (GUID) 16-byte code that identifies an interface to an object across all computers and networks. Such an identifier is unique because it contains a time stamp and a code based on the network address hard-wired on the host computer's LAN interface card. These identifiers are generated by a utility program.

GUID See Globally Unique Identifier.

ICO file (Application **Ico**n file) This is the graphic file for the application icon provided by the software vendor (for example, the Word icon for each Word file).

Jukebox database The Jukebox Server component that tracks user information and server resources for a given ***ASP***. All system and user information is stored in this database, including system configuration information, topology, logs, application information, application licenses, usage, and user account information.

See also database server.

Jukebox service This is a general term that refers to an Application Jukebox Server service.

See also Paging service, License service, and Admin service.

Kerberos A computer network authentication protocol which allows individuals communicating over a non-secure network to prove their identity to one another in a secure manner.

keytab file Used by Jukebox Server Enterprise Portal, this file contains pairs of ***Kerberos*** principals and DES-encrypted keys (derived from the Kerberos ***protocol***). The Keytab file can be used to log into Kerberos without being prompted for a password. The most common use of keytab files, however, is to allow scripts to authenticate to Kerberos without human interaction or store a password in a plaintext file.

license key A key used by software products to protect against piracy. Typically a license key includes a digital signature that locks to a specific computer to ensure the software cannot be executed on other computers.

License service A Jukebox Server component that manages ***application licenses*** and meters application usage. The primary functions are to grant, renew, and delete ***access tokens,*** record application usage, and help with server ***load balancing***.

load balancing The process of distributing processing and communications activity evenly across a computer network so that no single device is overwhelmed. Load balancing is especially important for networks for which it is difficult to predict the number of requests that will be issued to a server. If one server starts to get swamped, requests are forwarded to another server with more available capacity.

Load balancing refers to a server cluster sharing information requests equally across all of its active nodes. This can be done either statically, by tying clients directly to different back-end servers, or dynamically by having each client tied to a different back-end server controlled by software or a hardware device.

See also failover.

local paging The process used by Jukebox Player to page an appset from a file located on the Target PC. Local paging allows you to test several appset iterations quickly, without publishing the appset to Jukebox Server.

log file A file that stores messages generated by an application, service, or operating system. These messages are used to track the operations performed.

machine See server.

network segment A portion of a network separated by a Layer 3 switch.

node For Network topology, equipment situated at a point of branching of physical connections, or terminating a physical connection.

page A piece of a cloudified application that is fetched from the Jukebox Server system to Jukebox Player on users' PCs.

Paging service The Jukebox Server component that stores, manages, and pages cloudified applications to Jukebox Player on user's PCs. The Paging service adheres to restrictions established by the ***License service***.

See also License service.

persistent binding A setting that allows network switching hardware to constantly route traffic between two computers so that a ***session*** is maintained. With load-balancing switches, or Layer 3 switches, persistent binding settings facilitate this constant route instead of using the most efficient route.

Player cache The segment of disk storage on the Jukebox Player PC that temporarily stores pages that is fetched from an ASP's Application Jukebox Server.

See also cache.

prefetching The process of fetch application pages prior to their being explicitly requested by Jukebox Player, in anticipation of being needed by the application.

Stage-1 Prefetch - This group of application pages are those that are needed to start the application, as well as to perform typical actions for the first few minutes of using the program, such as opening files, saving files, etc.

Stage-2 Prefetch - This group of pre-paged data consists of application pages that are needed to accomplish other important tasks the user is likely to perform.

product license key The ***license key*** that is required in order to start the Jukebox Server system and register users.

project file The Jukebox Studio file (**.stw**) that stores all cloudifying configuration settings for an application that is being cloudified. Also called *STW file*.

project workspace The Jukebox Studio ***workspace*** for a project. It comprises, in particular, several views used to configure the settings of the project and manage its ***assets***.

See also workspace.

protocol Set of rules and conventions for sending information over a network. These rules govern the content, format, timing, sequencing, and error control of messages exchanged among network devices.

realm A database of user names and passwords that verifies valid users of a web service or application and displays a list of roles associated with each valid user.

Repository The Jukebox Server folder where all cloudified applications are stored. This folder must be accessible by all other Jukebox Server components.

scaled topology The installation of Jukebox Server components on multiple machines. For example, a multi-machine topology might include: ***Paging***, ***License***, and ***Admin services*** each installed on three separate machines, duplicates of these three machines for ***load balancing*** and ***failover***; a machine for the ***Jukebox database***; and (for Enterprise Portal or SaaS Portal) a machine for the Web Portal.

seats The total number of users that can concurrently run an application that is authorized by a particular application license.

Secure Sockets Layer (SSL) ***Protocol*** for secure network communications using a combination of public and secret key technology.

security Techniques for ensuring that data stored in a computer, or transferred from one computer to another, cannot be read or compromised. Most security measures involve data encryption and passwords. Data encryption is the translation of data into a form that is unintelligible without a deciphering mechanism. This typically requires a password, which is a secret word or phrase that a user enters to gain access to a particular program or system.

server A computer (or "machine") onto which Jukebox Server components or other system components are installed.

service For Application Jukebox Server, the sub-component of an ASP's Jukebox Server system that pages application ***pages*** to ***clients*** and provides other Jukebox Server functions.

See also Jukebox service and Windows service.

service interruption An interruption in the delivery of an application from an ASP's server. Jukebox Player handles and recovers from service interruptions that may occur during normal use. For example, Internet traffic can affect how data travels to and from networks.

session An open line of communication between a user and another computer, or between two computers. A session can remain open even if no communication is on-going. This allows for the communication to continue at a later time. Once a session has been evicted, then the communication will no longer be allowed and a new session must be established.

session key An encrypted key that is randomly generated to ensure the security of a communications ***session*** between a user and another computer or between two computers. Session keys are sometimes called ***symmetric keys***, because the same key is used for both encryption and decryption. A session key can contain information about the user that is communicating.

signed certificates Web sites use SSL certificates to secure communication with a web browser. A certificate is digitally signed by the creator and this signature is used by application software to validate the legitimacy of the certificate by others. Most web sites use a signed certificate issues from a certificate authority (CA), such as Thawte or Verisign.

Single Sign-On Application Jukebox Server Enterprise Portal uses the Windows ***authentication*** mechanism to achieve *Single Sign-On* with Windows Domain credentials. To properly use Single Sign-On, the Application Jukebox Server Enterprise Portal must be set up with a ***WINS*** name or a ***Fully Qualified Domain Name*** (***FQDN***) as part of the URL. This will allow users that have logged into a PC on the domain to access Application Jukebox Server Enterprise Portal without the need to additional authentication. Otherwise, the user will be prompted for authentication when an application is first launched.

SSL See Secure Sockets Layer.

STC file This file contains all the data pages of the application being cloudified. This file is the largest portion of the STP file.

STP file This compressed file contains all the ***appset*** files (.**stc**, **.aib**, **.tok**, etc.). Also called *appset file*. To uncompress an STP file, the **.stp** extension is changed to **.zip**.

strong password A password that satisfies these basic guidelines to protect against password-guessing system attacks:

- Include numbers, punctuation, and upper and lowercase letters in the password.
- Use a password having at least eight characters.
- Avoid a password based on repetition, dictionary words, letter or number sequences, user names, or biographical information such as names or dates.

STW file See project file.

symmetric key See session key.

system administrator For Application Jukebox Server, the person who manages the ASP's Jukebox Server server.

In general, the person responsible for setting up and managing domain controllers or local computers and their user and group accounts, assigning passwords and permissions, and helping users with networking issues.

target application An application that has been selected to be cloudified using Jukebox Studio.

templates Aliases that map to actual Windows folder paths. For example, WindowsDir maps to c:\windows.

TOK file See token file.

token file This file contains the security data and general information about the cloudified application. The token file extension is **.tok**. Also called *TOK file*.

token renewal frequency Frequency at which Jukebox Player must renew its ***access tokens*** in order to continue using cloudified applications. Currently, the frequency is set at 10 minutes.

Tomcat Apache Tomcat is a web container, or application server developed by the Apache Software Foundation (ASF). Tomcat implements the Java Servlet and the JavaServer Pages (JSP) specifications from Sun Microsystems, providing an environment for Java code to run in cooperation with a web server. It adds tools for configuration and management but can also be configured by editing configuration files that are normally XML-formatted. Tomcat includes its own internal HTTP server.

trigger A process or condition that causes an action to start, such as running a VB script.

virtual machine (VM) Software that mimics the performance of a hardware device, such as a program that allows applications written for an Intel processor to be run on a Motorola processor.

A VM is a computer that does not exist as a physical device, but is simulated by another computer.

virtualization The process that occurs after ***activation*** and ***prefetching*** of a cloudified application, which makes the application appear to the user's Windows system as if it were physically installed.

Web Portal (For Enterprise Portal and SaaS Portal systems) A web site that is integrated with Jukebox Player and Jukebox Server. Enterprise Portal is the web interface that functions as the front-end that an admin uses to configure and control all Jukebox services.

Web Server (For Enterprise Portal and SaaS Portal systems) The machine that contains the Web Portal.

Windows service An application that starts when Windows is booted and runs in the background as long as Windows is running. Windows services can have special privileges that regular applications do not. For instance, on a limited user account a service could access certain parts of the Windows Registry that a regular application could not access. Most Windows services do not have a user interface and are loaded when Windows starts.

WINS (Windows Internet Name Service) Microsoft's implementation of NetBIOS Name Service (NBNS), a name server and service for NetBIOS computer names. Effectively, WINS is to NetBIOS names what DNS is to domain names - a central mapping of host names to network addresses. Like DNS, it is broken into two parts: (a) a Server Service that manages the encoded Jet Database, server to server replication, service requests (query & registration, renewals, de-registration), and conflicts; and (b) a TCP/IP Client component which manages the client's registration, renewal of names and takes care of queries.

workspace The part of the Jukebox Studio User Interface (UI) that consists of the main window and all UI components within it (menu bar, toolbar, status bar, the various panes and views, etc.). The UI components that appear in the workspace vary based on the type of Jukebox Studio file currently being worked on.

See also project workspace.

Index

Symbols

.aib, 19
  See also AIB file
.cae, 19
  See also CAE file
.html, 19
  See also HTML file
.ico, 19
  See also ICO file
.stc, 19
  See also STC file
.stp, 19
  See also STP file
.stw, 6
  See also STW file
.tok, 19
  See also TOK file

A

access token, 173
activation, 173
active
  view, 10
  workspace, 10
adding
  Configurable AppEvents
  environment variables, 90
  fonts, 93
  registry keys, 84
  registry values, 87
  root folders, 75
  services and drivers, 103
  startup items, 98
  sub-folders, 76
Admin service, 173
admin. *See* system administrator
AIB file, 9, 13, 19, 173
  modifying file settings, 119
  modifying registry settings, 122
  opening, 118
AIB view, 13
  See also views
AIB workspace, 9, 13
  See also workspace
AppEvent. *See* Configurable AppEvent
application, 173
  extracting, 160
Application Jukebox Studio, 1, 5
application license, 173
Application Service Provider, 173
appset, 5, 173
  extracting an application from, 160
  patching, 157, 158
  See also cloudified application
appset designer, 2, 174
appset file, 19, 180
  extracting. *See* appset file, uncompressing
  opening, 118
  uncompressing, 19, 118, 180
  See also STP file, 19
artifacts, 71, 174
ASP. *See* Application Service Provider
asset, 5, 6, 9, 174
authentication, 174

B

buffering. *See* prefetching

C

cache, 69, 174

size adjustment, 52
CAE file, 19, 174
CAE. *See* Configurable AppEvent
capturing, 6
Caution icon, 3
CDData, 73
CGI script, 174
clean PC, 2, 22, 174
client, 174
cloudified application, 6, 175
See also appset
cloudifying, 6, 45, 53, 175
environment, 2
environment back-up, 53
performance, 22
See also cloudifying phases
cloudifying phases, 15
(Phase 1) preparing the environment, 21
(Phase 2) capturing the application installation, 27
(Phase 3) cloudifying the application, 43
(Phase 4) testing and optimizing the appset, 47
cloudifying project. *See* project
cloudpaging, 175
command line, 38
setting, 82
common area, 9
See also UI
compatibility settings, 30
advanced text services, 42
desktop composition, 42
display scaling on high DPI settings, 42
reduced color mode, 41
run as administrator, 42
screen colors, 41
screen resolution, 41
SkyDrive files accessibility, 42
visual themes, 41
compression, 8, 30
setting, 52
Configurable AppEvent, 58, 173, 175
adding, 58
Configurable AppEvent file, 19
See also CAE file, 19
configuration information, 5, 6
conventions
icons, 3
typographic, 3
core concepts, 7

D

database server, 175
See also Jukebox database
deleting
environment variables, 91
files and folders, 81
fonts, 96
registry keys and values, 88
services and drivers, 107
startup items, 101
dialogs, 14
modal, 14
modeless, 14
disposition layers, 78, 120, 124, 175
(Layer 1) installed-permanent, 7
(Layer 2) installed-temporary, 7
(Layer 3) virtual-integrated, 7
(Layer 4) virtual-isolated, 7
global changes, 113
See also virtualization
Documents and Settings folder, 73
drivers. *See* services and drivers
DSN file, 175

E

editing
environment variables, 91
file properties, 80
files, 80
folder properties, 77
fonts, 95
registry key disposition, 86
registry values, 88
services and drivers, 105
startup items, 99
encryption, 8, 31
setting, 52
End User License Agreement, 19, 32, 175
environment variables
adding, 90
deleting, 91
editing, 91
excluding, 91

virtualization, 62
Environment view, 11, 89–91
See also views
EULA. *See* End User License Agreement
excluding
environment variables, 91
files and folders, 81
fonts, 96
registry keys, 88
services and drivers, 108
startup items, 101
extracting
application, 160
application icon, 32
appset, 118

F

failover, 176
file associations
not working. *See* problem solutions
file extensions
.cae. *See* .cae
.html. *See* .html
.ico. *See* .ico
.stc. *See* .stc
.stp. *See* .stp
.stw. *See* .stw
.tok. *See* .tok
file types
AIB file, 9
appset file. *See* .stp, 19
project file, 6
token file, 9
files
adding to a folder, 79
deleting, 81
editing, 80
editing properties, 80
excluding, 81
finding, 109
Files view, 11, 72–82
See also views
Find Results window, 14, 110, 110–??, 110, ??–111
finding items, 109–110
clear list, 111
editing properties, 111
j ump to an item, 111
remove item from list, 111
firewall, 176
fixed license, 176
floating license, 176
folders
adding files, 79
adding root folders, 75
deleting, 81
editing folder properties, 77
excluding, 81
finding, 109
merged, 73
renaming, 77
root, 73
templates, 73
types, 73–74
working folder, 78
fonts
adding, 93
deleting, 96
editing, 95
excluding, 96
file properties, 95
injection, 63
virtualization, 63
Fonts view, 11, 92–96
See also views
FQDN. *See* Fully Qualified Domain Name
Fully Qualified Domain Name, 176

G

game ratings, 32
Global Disposition Settings dialog, 113
Globally Unique Identifier, 176
GUI. *See* UI
GUID. *See* Globally Unique Identifier

H

help, 3
HTML file, 19

I

ICO file, 19, 176
icon
extracting (for application), 32
icon file, 19

See also ICO file, 19
icons
conventions, 3
injection, 7
fonts, 63
installation, 23

J

Jukebox database, 176
Jukebox service, 176
Admin, 173
License, 177
Paging, 177

K

Kerberos, 177
keytab file, 177

L

license key, 177
License service, 177
load balancing, 177
local paging, 48, 177
log file, 177

M

machine. *See* server
main window, 9–??, 9, ??–9
See also UI
menu bar, 9
See also UI
merged folders, 73
setting, 79
modal. *See* dialogs
modeless. *See* dialogs

N

network segment, 177
node, 177
Note icon, 3

O

optimizing
appset, 49

P

paging
local, 48, 177
settings, 56
See also cloudpaging
Paging service, 177
patched appset
modifying, 160
patching
appset, 158
diagram, 157
performance
settings, 69
persistent binding, 178
Player cache, 178
See also cache
plug-ins, 27
prefetching, 7, 178
creating a prefetch file, 49
stages, 7
problem solutions, 147
application copy protection, 154
application fails to launch, 149
approaches to uncommon problems, 154
file associations do not work, 149
fonts do not appear, 150
missing shortcuts, 148
printing problems, 149
roaming and folder redirection issues, 150
Windows Installer pop-up dialogs appear, 152
Windows service does not start after launching, 152
product license key, 178
project, 5, 6, 11
configuration, 5, 6
project file, 6, 11, 178
See also project
See also project workspace
See also STW file
Project view, 11
See also views
project workspace, 11, 178
See also workspace
protocol, 178
publishing an appset, 53

R

realm, 178
registry
  adding keys, 84
  adding subkeys, 84
  adding values, 87
  changing the disposition of a key, 86
  deleting keys and values, 88
  editing values, 88
  excluding keys, 88
  finding keys, 109
  finding values, 109
  keys, 84
  renaming keys, 86
  subkeys, 84
  values, 87
Registry view, 11, 83–89
  See also views
regular expressions, 112
renaming
  folders, 77
  registry keys, 86
repository, 178
root folders, 73
  adding, 75
Run items. *See* startup items
Run Keys, 97

S

sandboxing
  settings, 63
search type, 111–112
  regular expressions, 112
  wildcards, 111
searching for items. *See* finding items
seats, 178
Secure Sockets Layer, 178
security, 179
  override, 66
  settings, 66, 78
server, 179
service, 179
service interruption, 179
Services & Drivers view, 11, 102–108
  See also views
services and drivers
  adding, 103
  deleting, 107
  editing, 105
  excluding, 108
  image file properties, 104
session, 179
session key, 179
shortcuts
  missing. *See* problem solutions
signed certificates, 179
Single Sign-On, 179
SSL. *See* Secure Sockets Layer
stages. *See* prefetching
Start Menu Items, 98
startup items
  adding, 98
  command line properties, 100
  deleting, 101
  editing, 99
  excluding, 101
  types, 97
Startup view, 11, 97–101
  See also views
status bar, 9
  See also UI
STC file, 19, 179
STP file, 19, 180
  uncompressing, 19, 118, 180
strong password, 180
Studio project. *See* project
STW file, 6, 178
  See also project file
sub-folders
  adding, 76
support, 3
symmetric key. *See* session key
system administrator, 180
system requirements, 32

T

target application, 180
technical requirements, 21
templates, 73, 180

template mapping samples, 163

testing
appsets, 48
See also problem solutions

TOK file, 19

TOK file. *See* token file

token file, 9, 13, 19, 180
modifying a value, 130
modifying game rating, 132
modifying OS compatibility settings, 133
See also TOK file, 19

token renewal frequency, 180

Token view, 13
See also views

token workspace, 9, 13
See also workspace

Tomcat, 180

toolbar, 9
See also UI

trigger, 180

troubleshooting. *See* problem solutions

typographic conventions, 3

U

UI, 9–14
common area, 9
main window, 9
menu bar, 9
status bar, 9
toolbar, 9
views, 9
workspace, 9
See also dialogs
See also views, 9
See also workspace

uncompressing
appset, 118

V

views, 9, 10–??, 10, ??–13
AIB view, 13
Environment, 11, 89
Files, 11, 72
Fonts, 11, 92
Project, 11
Registry, 11, 83
Services & Drivers, 11, 102
Startup, 11, 97
Token, 13
See also UI

virtual machine, 23, 180

virtualization, 7, 180
environment variables, 62
fonts, 63
settings, 62

W

Web Portal, 181

Web Server, 181

wildcards, 111

Windows service, 152, 181

WINS, 181

working folder, 39
setting, 78

workspace, 9–??, 9, ??–13, 181
See also AIB workspace
See also project workspace
See also token workspace
See also UI